Automation of the analysis of comments in the University of Lleida surveys

David Sánchez Marín

Master in Computers Engineering

Project director: Roberto García González

Departament: Computing and Industrial Engineering

Polytechnic School, University of Lleida

Campus of Cappont. EPS Building. Office 3.15

606 42 40 13 / dsm9@alumnes.udl.cat

Summary

The aim of this Master's Final Project is to develop a tool to automate the process of consultation, review and **analysis of comments** in the satisfaction surveys performed by the University of Lleida, in particular by the Quality and Teaching Planning unit. It will be oriented to **Natural Language Processing (NLP)**.

This tool will provide the following main features:

- Consultation of the initial status of the comments and assessment of the surveys to be treated.

- Use of Artificial Intelligence technology in the treatment of comments with the aim of detecting conflictive or inadmissible comments and other types of issue.

- Visualization of the result of the revision with the possibility of manually retouching the final sentences.

Other objectives will be:

- Transversal use of the technologies presented throughout the Master.

- Exploring the state of the art in Artificial Intelligence with regard to text analysis. The selected NLP tool is *Spacy [1]*.

- Test of frameworks to develop web applications using the Python language. The programming framework is *Django [2]*.

- Use of an agile development methodology in the development of the project. In this case is ***Behaviour Driven Development (BDD) [3]***.

1. Introducció

The University of Lleida conducts a wide range of types of surveys, aimed at students, faculty, administrative staff, interns in companies, … As a general rule, all surveys include a set of closed-ended questions, where questions should be rated from 1 to 5. But they also include in almost all cases free text questions for the respondent to express their comments on the teacher, the subject, the degree, …

The problem is that a percentage of responses include foul-sounding and offensive expressions that are not considered appropriate to use and disseminate. To solve this problem the staff of the Quality unit performs a manual revision of the comments eliminating derogatory sentences or replacing them with expressions with the same meaning as expressed more correctly. Other types of issues are reviewed, for example, professors that haven't taught the group, comments related to the professor answered in the subject part, etc.

Every year nearly **75.000 surveys** are generated, of which 31.000 have responses. These responses include **14.000 free comments**. Some action was taken in 350 (2.5%) of them.

It isn't a large number, but in order to detect these cases, all the comments have to be reviewed. If the tool to be developed performs the task of detection would be an important help by focusing the effort on comments that require some intervention.

2. Development of the project

The implementation of the project was done in successive iterations:

- Iteration 1: Definition of the user interface. It includes the description of the original application, *LimeSurvey [4]*, the definition of the new database and the features and scenarios of the project, following BDD methodology.

- Iteration 2: Loading and transformation of the sample files using *Jupyter* notebooks and *Pandas*.

- Iteration 3: Analysis and improvement of language detection. It used a mix of the library *LanguageDetect* of *Spacy* and *PyCld2*.

- Iteration 4: Development of a *Spacy* development model to detect the issue type 'Professor hasn't taught this group', using the sample comments labeled by the users and the class *TextCategorizer*.

- Iteration 5: Development of a new detection alternative of this issue, using *Part-of-speech* and *Dependency parsing*. Comparison between the two models.

- Iteration 6: Implementation of the user interface developed with the framework *Django*. It includes the basic features but using sample information.

- Iteration 7: Importation of surveys and comments from the source application *LimeSurvey*. It required solving a lot of challenges such as managing multiple databases, accessing legacy databases, using multiple key tables, etc.

- Iteration 8: Integration on the NLP Spacy model of issue type 1 in the user interface created with Django.

- Iteration 9: Creation of a new Spacy NLP model corresponding to the issue type 'Problematic comments'. It includes the analysis of different alternatives.

- Iteration 10: Integration of the new Spacy NLP model 2 in the user interface, creation of a Celery task to execute the process asynchronously and programming of the feedback to the user.

3. General results

As a result of the project, we have an application capable of import surveys and comments from the original Lime database, process it with NLP to classify two of the possible issue types, show the user the list of campaigns, the list of surveys and comments, and allow the edition of these comments.

The main screen is the list of surveys and comments where the user has the processing comments button, the result of the language detection, and the result of the issue type classification.

Figure 1 List of surveys and comments screen

Results of the **language detection** process, it has been obtained the largest part of the comment are in Catalan (74%), followed by Spanish (16,5%) and a little part in English (3%). Only 6,4% aren't identified, that is a 93,6% of detection capacity.

The evaluation of the **Spacy NLP model 1** had obtained a precision of 89% in Catalan and 91% in Spanish. Model detects positive cases well, but not so many false cases.
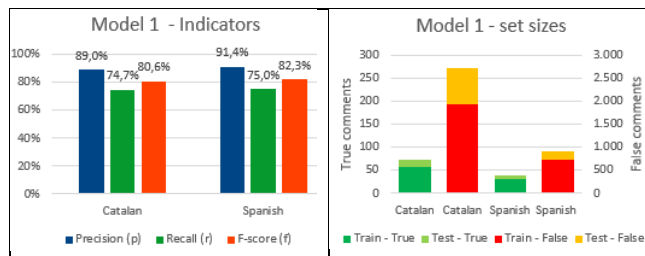

Figure 2 Model 1 - Indicators and set sizes

In the construction of the training a test set all the positive cases had been chosen and a selection of the false negatives, because there are few positive cases.

There is insufficient data in English to train a model.

In the implementation of **Spacy NLP model 2** the key was the selection of the parameters to train the model. It was considered the strategy (use of full comments or individual sentences), the pretrained model (small or large) and the set sizes. The configuration with better performance was the use of full comments, large pretrained models and small training set with 40% of true cases.
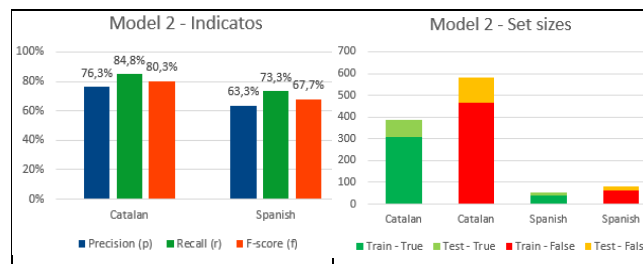
4. Conclusions

At the end of this project, it had been developed and fully functional application that allows user import campaigns, surveys and comments, detect language of the comments and classify them in two of the desired issue types. It also permits showing the result of the importation and classification and editing manually the comments if needed.

The development has followed the Agile methodology Behaviour Driven Development.

The implementation of the user interface with Django has made it possible to test this powerful Python oriented framework.

And finally, it has been an opportunity to explore the use of the Spacy NLP library in the analysis and classification of texts.

It was not possible to implement all the originally proposed functionalities but the core of the application. This is an occasion for future developments.

5. Acknowledgements

References

[1] Industrial-Strength Natural Language Processing, ExplosionAI GmbH
https://spacy.io

[2] Django documentation, Django Software Foundation
https://docs.djangoproject.com/en/3.1/

[3] What is BDD (Behavior Driven Development)?, Agile Alliance
https://www.agilealliance.org/glossary/bdd/

[4] Universidades - LimeSurvey - Easy online survey tool, LimeSurvey GmbH
https://www.limesurvey.org/es/soluciones/universidades