Cardiff University
Computer Science and Informatics

CM3203 – One Semester Individual Project – 40 Credits

IDIOM SEARCH ENGINE

Initial Plan

Author: Callum Hughes
Supervisor: Irena Spasic
Moderator: Martin Caminada

1. Project Description

Idioms are phrases (i.e. groups of words) whose meaning may not be deducible from those of the individual words (e.g. "bury the hatchet" = "end a quarrel or conflict and become friendly"). One difficulty in finding idioms in text is the fact that they can vary, so a simple string search is not appropriate. For example, searching for "bury the hatchet" would miss this idiom in the following two sentences:

‘Christmas looks to be a time for burying the hatchet or exhuming it for re-examination.’

‘From the look of things, the hatchet has been long buried.’

The aim of this project is to implement an idiom search engine, which would take an idiom as input and find all of its occurrences in a corpus of text.

This would provide linguists with a valuable tool for analysing how a particular idiom might be used within language. In comparison to existing search tools when searching for idioms [1], the idiom search engine could allow greater accuracy of results whilst still using a straightforward input from the user.

2. Project Aims and Objectives

In order to recognise a given idiom in a given text, the project will aim to implement a method for automatically detecting idioms provided through a free text field input. This allows the possibility for large scale coverage of given idioms by the user to be recognised, in comparison to a finite set of manually selected idioms. A previous study detailing methods for the automatic encoding of local grammars [2] could be used to support the development and understanding required to recognise idioms automatically, either by implementing the pattern generation approach exactly or by supporting a new method using aspects of the approach.

The project can analyse the results of the final implementation against a gold-standard set of manually defined lexico-semantic pattern matching rules developed within a previous study [3] and available from the following URL: http://www.cs.cf.ac.uk/idioment

When comparing the final implementation to the gold-standard approach (using the test dataset of 500 sentences), it should be

expected to perform at an equivalent level to the previous studies implementation against the same comparison [2].

The web application used to support the search feature will also adhere to the most recent OWASP top 10 web application security standards [4].

2.1 KEY MILESTONES

- Reach an agreement for specific functional and non-functional requirements with the client.
- Research and evaluate an efficient and accurate approach towards automatically recognising an idiom in text.
- Develop a 'walking skeleton'[5] for the application back-end (API).
- Develop a 'walking skeleton' for the application front-end.
- Implement a compatible database for the system and collect a corpus of searchable text.
- Implement and test the API for idiom recognition and searching.
- Design and program the user interface and develop compatibility with the search API.
- Test and evaluate the complete implementation against the set-out requirements.

3. WORK PLAN

The following work plan is prepared to set a baseline for how each part of the project will be worked on in sequence throughout the time period (waterfall approach). This is also used to identify some of the potential challenges yet to be faced and predict the amount of time to account towards them. It's expected that errors in predictions and unknown challenges will result in changes to the work plan over the course of development. Time slots are approached with reasonable pessimism in order to allow for any stages that over-run or features yet to be negotiated, to be accounted for within other weeks.

Week 1: A specific set of functional and non-functional requirements for the application with the client should be developed. For example, text input size limitations, search history features, expected processing time, website availability and language limitations have yet to be agreed with the client. Therefore, a meeting should be carried out to specify and establish this functionality before the beginning of development.

Next suitable technologies, frameworks and the overall architecture needs to be evaluated and components selected for use in the application. A cloud service provider also needs to be selected, allowing both the front-end and back-end of the application to be developed into a ‘walking skeleton’ complete with some basic unit tests. A suitable corpus of text relevant to the client needs to be collected and stored in a database for searching over.

Weeks 2, 3 and 4: Work on the API for processing idiom recognition within text can begin. This will first need to consider the endpoint request and response structure and the handling of requests within the application. Relevant packages/libraries need to be chosen in order to provide some of the natural language processing functionality necessary to complete the implementation. Previous studies and approaches can be referenced in order to achieve accurate results.

Week 5: The idiom recognition API can be tested against test sentences to evaluate performance using precision and recall statistics. User stories, interface design and component libraries can be selected in preparation for user interface development.

Week 6: The user interface pages can be developed along with compatibility with the back-end API.

Week 7: Domain name and certificates can be acquired. Compliance with OWASP top 10 security principles can be evaluated.

Week 8: Final testing and evaluation of requirements against the application can be made as well as collect any necessary data for performance measuring.

Weeks 9, 10 and 11: Summarise results and write the final report for the project.

REFERENCES

[1] "British National Corpus (BNC)", English-corpora.org, 2020. [Online]. Available: https://www.english-corpora.org/bnc/. [Accessed: 30- Jan- 2020].

[2] I. Spasic, L. Williams and A. Buerki, "Idiom—based features in sentiment analysis: Cutting the Gordian knot", IEEE Transactions on Affective Computing, pp. 1-1, 2019. Available: 10.1109/taffc.2017.2777842.

[3] L. Williams, C. Bannister, M. Arribas-Ayllon, A. Preece and I. Spasić, "The role of idioms in sentiment analysis", Expert Systems with Applications, vol. 42, no. 21, pp. 7375-7385, 2015. Available: 10.1016/j.eswa.2015.05.039.

[4] "OWASP Top Ten", Owasp.org, 2017. [Online]. Available: https://owasp.org/www-project-top-ten/. [Accessed: 30- Jan- 2020].

[5] H. Kousar and K. Kumar, "Walking Skeleton Strategy in a Test Driven Development", International Journal of Scientific and Research Publications, vol. 4, no. 4, pp. 141-148, 2014. Available: 10.1.1.432.9280.