Analysis and visualisation of opinion diffusion

Project Plan

Chris Coe
c.coe.12@aberdeen.ac.uk

Department of Computing Science,
University of Aberdeen, Aberdeen AB24 3UE, UK

Introduction

The ability to automatically analyse texts and detect the topics and sentiment within has been a longstanding goal in computer science. In addition, finding a way to adequately summarise and visualise the data from such analysis is important if it is to be useful. This project will seek to produce a piece of software that can analyse a large set of documents (e.g. a set of reviews for a product) and extract common topics and sentiment information from these, and visualise them in a way that makes interpretation of the results easier.

This will be useful to, for example, people who need to make sense of a large number of individual documents that would be too laborious to read and classify individually. For example, an employee at a company sifting through hundreds of feedback forms or reviews, and having to keep track of recurring topics.

Goals

The goal of the project will be to produce a piece of software that can perform the task mentioned above: analyse a set of documents and extract from these some common topics or themes, along with sentiment information for these, and visualise them in a stream graph[1] style. Additionally, it would be useful to try to detect the important moments in the series, such as topic or sentiment changes (e.g. sentiment changes coinciding with the release of a new version of the software). The goal of detecting the key moments depends on the first goal being achieved, apart from the actual visualisation of the data, so this could be considered a secondary objective.

There are a few existing natural language processing packages available, such as Stanford CoreNLP, which will help with the processing of the text. There is also a multitude of techniques available for processing sentiment information, so choosing an appropriate one to implement will be important. Graphing packages for creating stream graphs are also available, so these could be incorporated as well.

Methodology

Several steps will be taken in order to achieve the goals of the project in good time:

- Related and other relevant papers and previous work will be searched for and studied, in order to gain insight into the state-of-the-art approaches currently being used, and potential pitfalls to avoid. Topics of particular interest will be sentiment and topic modelling, and data visualisation techniques.

- A programming language in which to write the software will be chosen, after considering which packages are available for each, and how well they can interact with other components.
- A data set of documents on which to train the software will be collected and annotated for sentiment and topic. This will be used to both train and test the software's sentiment and topic analysis abilities.
- A prototype will be developed that can visualise data. Early prototypes may just work on pre-classified data, while later ones will build in the analysis stages as well.
- Once the software is tuned and improved as much as possible, it will be tested for accuracy against the human-annotated set to see how well it performs.
- A report on the project will be written.

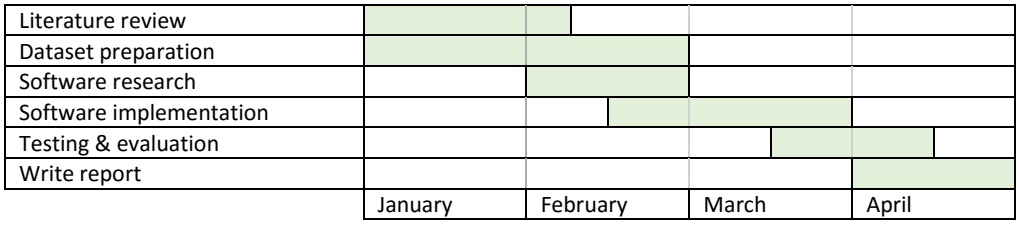
Resources Required

This project should only require a PC to run. Depending on the techniques used, dataset training tasks may require large amounts of RAM, so a high-end computer will likely be needed for this stage.

Risk Assessment

The only major risk for this project is of time running out for achieving the goals. However, careful planning should ensure that the project moves along smoothly. If time is running short, the secondary goal of detecting significant moments could be dropped in order to allow more time for the more fundamental tasks.

Timetable



[1] Byron, Lee, and Martin Wattenberg. "Stacked graphs–geometry & aesthetics." *Visualization and Computer Graphics, IEEE Transactions on*14.6 (2008): 1245-1252.