Toward Human-Like Summaries Generated from Heterogeneous Software Artefacts

Mahfouth Alghamdi, Christoph Treude, and Markus Wagner

School of Computer Science

THE UNIVERSITY of ADELAIDE

Summarisation Everywhere…

- News headlines.
- Abstract of research papers.

THE WALL STREET JOURNAL.

THURSDAY, SEPTEMBER 18, 2008 - VOL. CCLII NO. 67

Mounting Fears Shake World Markets As Banking Giants Rush to Raise Capital

Morgan Stanley in Talks With Wachovia, Others

High Anxiety, Low Returns

What's News—

Bad Bets and Cash Crunch Pushed Ailing AIG to Brink

Worst Crisis Since '30s, With No End

International Journal of Advanced Research and Development

International Journal of Advanced Research and Development
ISSN: 2455-4030, Impact Factor: RJIF 5.24
www.advancedjournal.com
Volume 2; Issue 4; July 2017; Page No. 29-38

A survey on: Extractive text document summarization techniques

Chandra Shekhar Yadav, Rakesh Kumar, Prem Shankar Singh Aydav, Harendra Pratap Singh
School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

Abstract
In modern word, popularity of the internet is growing day by day so, volume of data is growing exponentially on the web and a variety of information services increased therefore, to obtain any desired information became a challenging task. Text summarization may be a feasible and powerful to handle such type of problems. Even text summarization can help to the government for "Good Governance". In fact, e-governance is going to be a new approach of any government for Good governance. Summarized information also helps in many ways like in emergency decision support, policymaking, and government routine for government as well as for civil servants. For example Indian government recently launched a web portal www.mygov.in for all people, to give their contribution/views for policy making (like for Digital India, Green India, Skill development etc), to share views about different issues like the recent United Nations declared 21st June at 'International Day of Yoga'. Such type of initiative can help in better policymaking and this is possible only through Text Document Summarization.

Keywords: single document summarization, multi document summarization

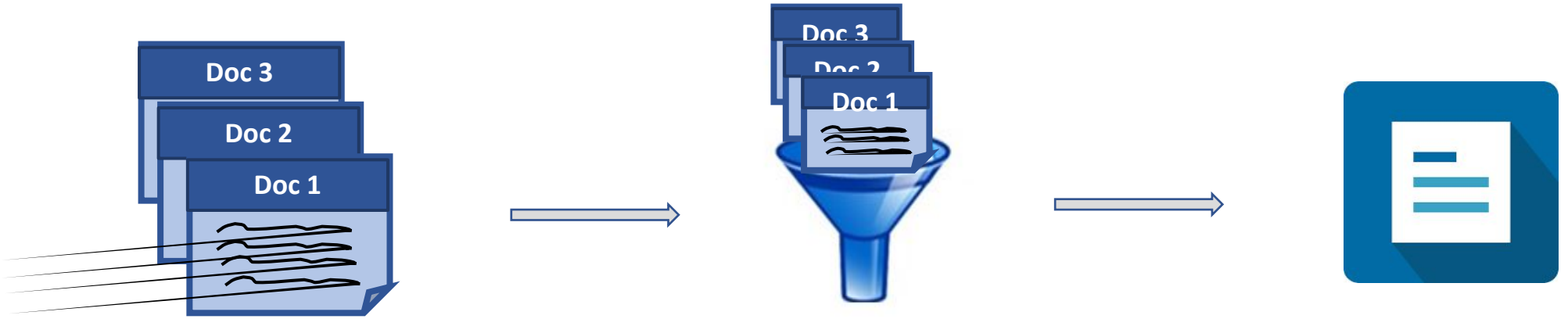
1. Introduction
Nowadays, summarization of text document have very effective role in information retrieval (IR) because it presents a huge set of information in summarized form by considering the important and relevant sentences. In brief, according to M. Ramiz summarization in [1] is considered as three steps procedure (1) Text analysis (2) Summary representation known as Transformation, and (3) Synthesis- Generation of relevant summary. The first phase is the analysis phase, In this phase we have a task to analyze the given text document and to select some salient features. After analysis phase in transformation phase, transform the analyzed text into a summary representation based on selected features and the final phase which can be called synthesis (or sometime generation step) in which task to already represented summary are taken to produce more appropriate summary according to user need. Eduard Hovy and Chin-Yew Lin [2] introduced SUMMARIST system to create a racy automated Text Summarization system, three phase working of the system can be understood with equation as "Summarization = Topic Identification + Interpretation + Generation". The compression rate also plays an important role in summarization and effect of this (compression rate) can be

1.1 Summarization
Radev *et al*. in [6] has define a text summary as "a text that is produced from one or more texts, hat conveys important information in the original texts, and that is no longer than half of the original text and usually significant less than that".

1.2 Type of Summary
Basically type of summary can depends on (1) number of documents. (2) approach applied, and (3) user needs. According to number of document summarization may be single document or multi document, if we wants to classify according to approached then it may be Extractive, Abstractive (human like summary), if we wants to classify according to user need then it may be informative (query specific)or generic summarization.
Number of Documents: Single Document or Multi Document.
Technique: Extraction or Abstraction.
Detail: Indicative or Informative.
Content required by user: Generalized or Query-based.
Approach: Either Shallow or Deep, Domain specific, Template based, Statically or Soft computing.

Automatic Text Summarisation.

- Goal of the automatic text summarisation is “to take an information source, extract content from it, and present the most important content to the user in a condensed form” [1]
- Applications
  - Social networks
  - Software engineering data



Original artefacts text

Summariser

Summarised output

[1] Maybury, Mark T., and Inderjeet Mani. "Automatic summarization.“, In *ACL Meeting Toulouse,*2001

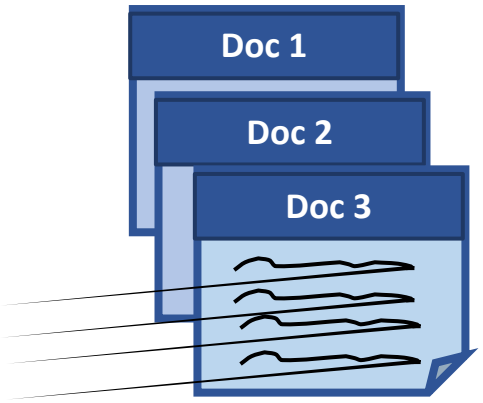
Approaches for Automatic Text Summarization.

Input

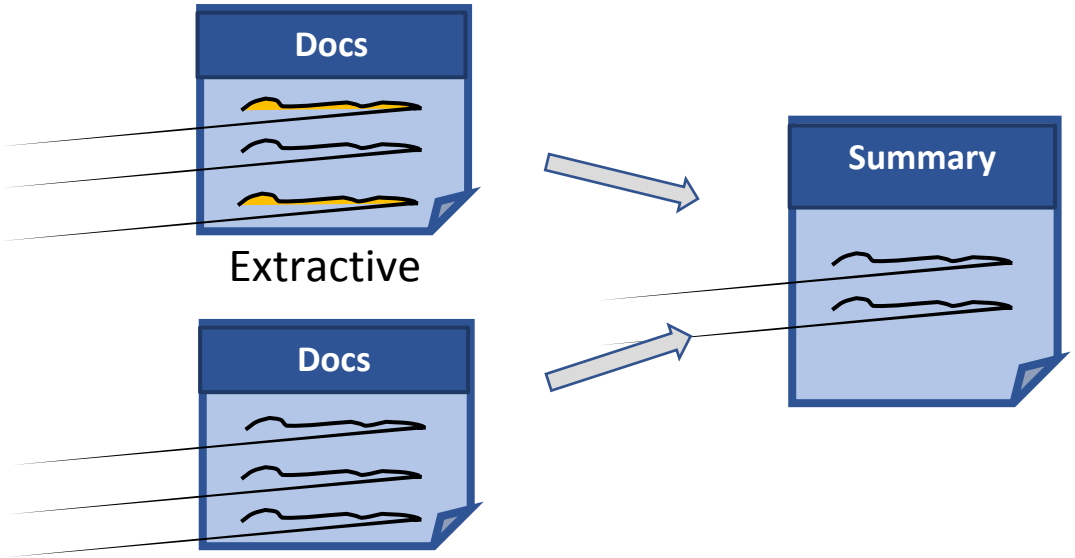
- Single-document
- Multi-document

Output

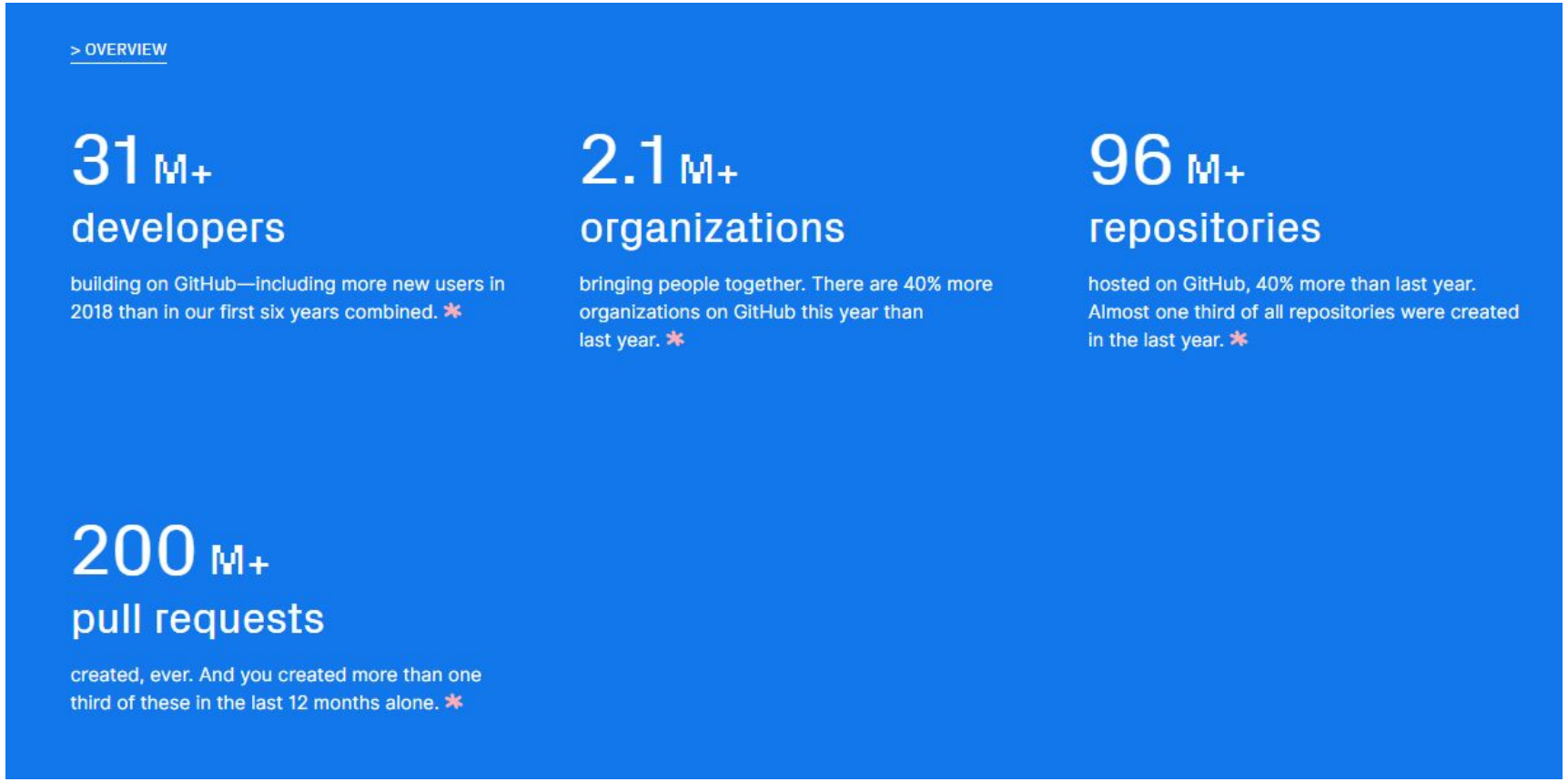
- Extractive
- Abstractive



Multi-document



Information Overload: GitHub.



Facts about GitHub's statistics (from Oct 1st, 2017 to Sept 30th 2018)

https://octoverse.github.com/

Why Should We Summarise Software Engineering Data?

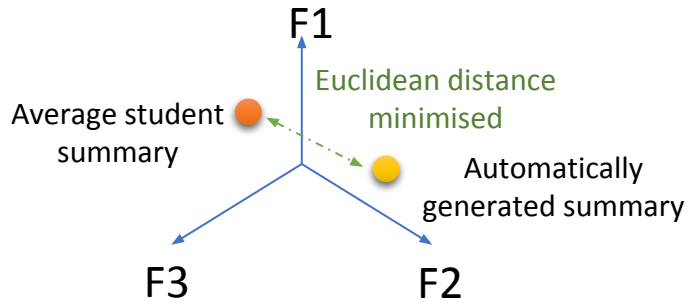
- To gain a comprehensive understanding of the contribution and productivity of an individual developer [2].
  - Help the new developer to integrate quickly into existing software project development.
  - Help the manager to improve the productivity of the developer.

[2] http://ctreude.ca/2018/01/03/automatically-summarising-and-measuring-software-development-activity/

Generating Summaries From Software Artefacts: *A Challenge Task.*

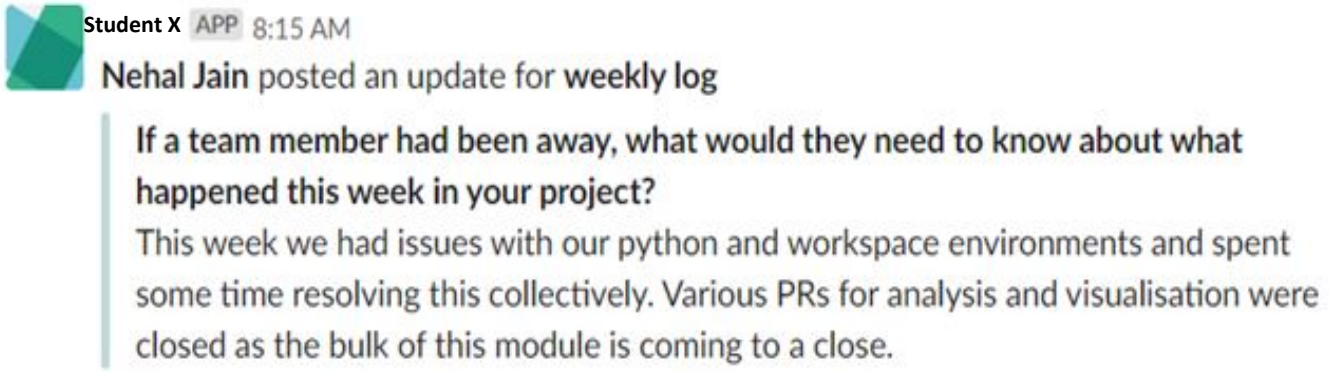
- Summaries produced from multi-document which contains heterogeneous software artefacts.

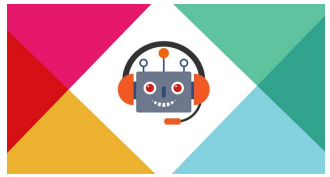
- First step toward our ultimate goal:
  - Understanding the characteristics of the students' summaries.
    - can help us to solve a subset selection problems
  - The characterization of these summaries will guide the search in at least two ways:
    - In a single-objective formulation.
    - In a many-objective optimisation problem.



Characteristics of Students' Summaries.

- We collected a total of 545 human-written summaries produced on a weekly basis by 53 students from 15 GitHub projects
- All the summaries were analysed using 27 features related to readability metrics, lexical features, and information theoretic entropy.





- The students' summaries were grouped by courses, weeks, and teams.

Characteristics of Students' Summaries (*continued*).

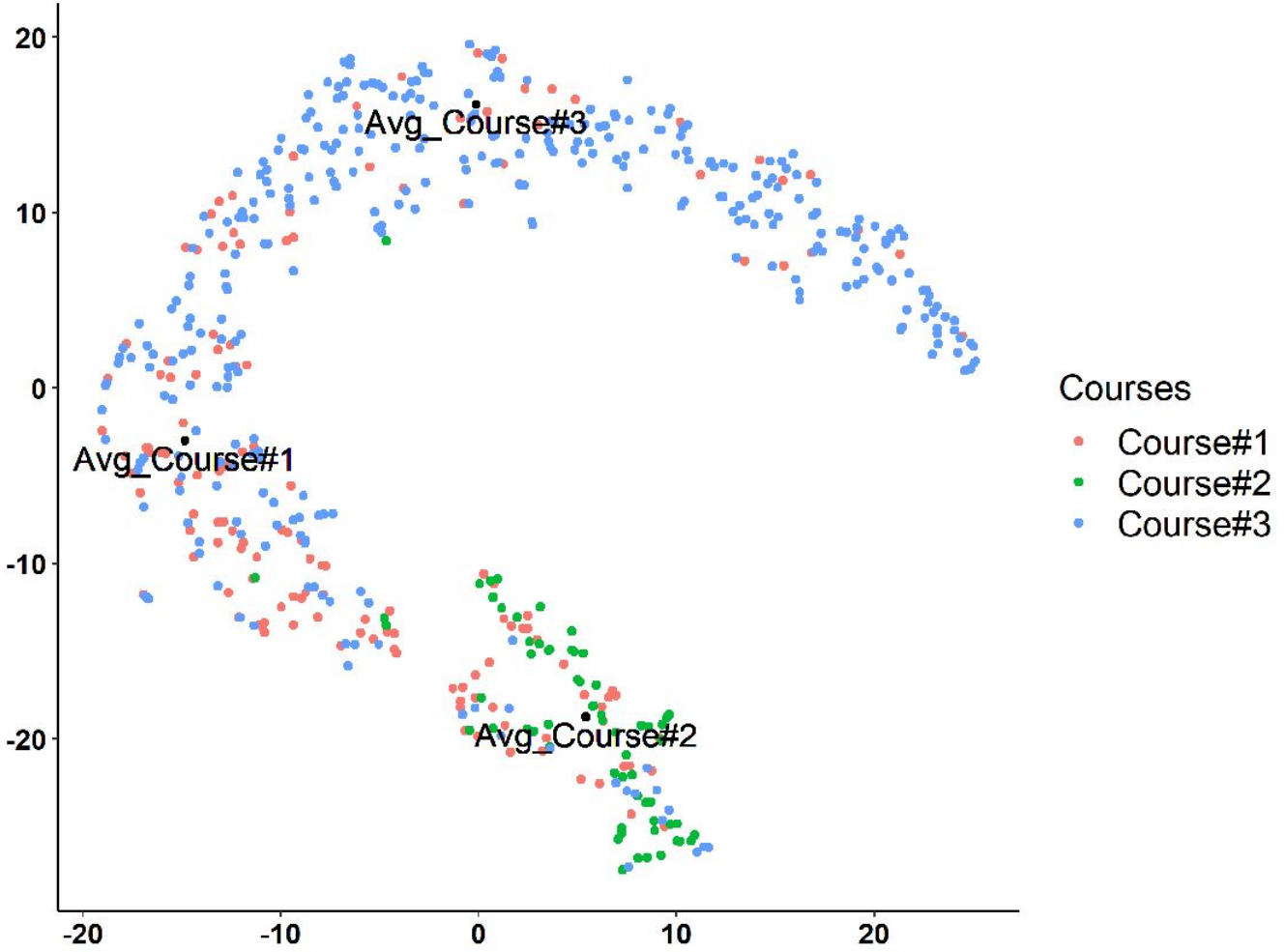
- For inspecting and visualizing our 27-dimensional characterisation, t-distributed Stochastic Neighbour Embedding (t-SNE) [3] was used.

- To facilitate the interpretation, we added (before employing t-SNE) to each grouping the respective Euclidean average as each group's centre.

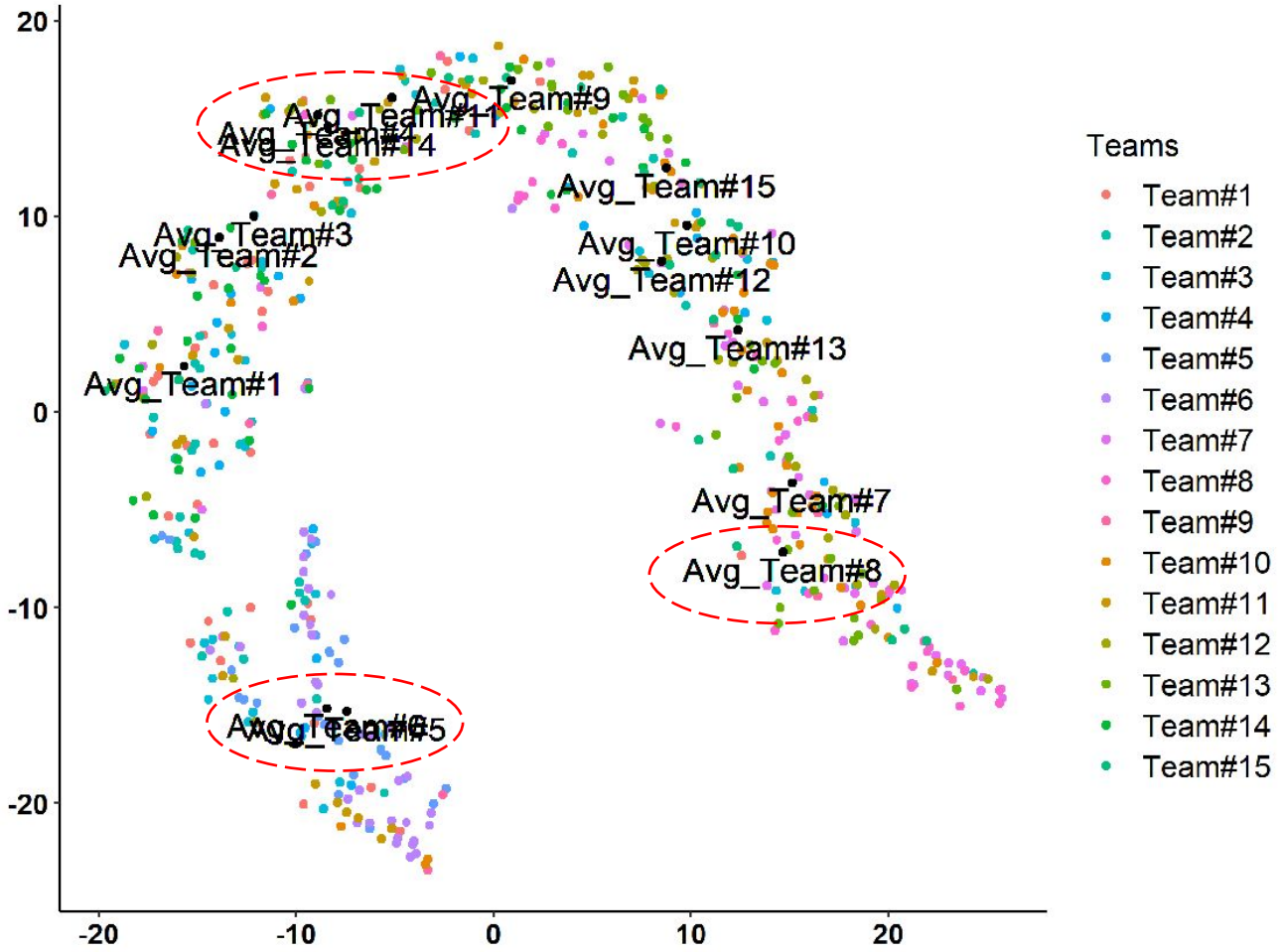
[3] L. v. d. Maaten and G. Hinton. 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9 (2008), 2579–2605

Results: characteristics of the student summaries based on text features grouped by courses.

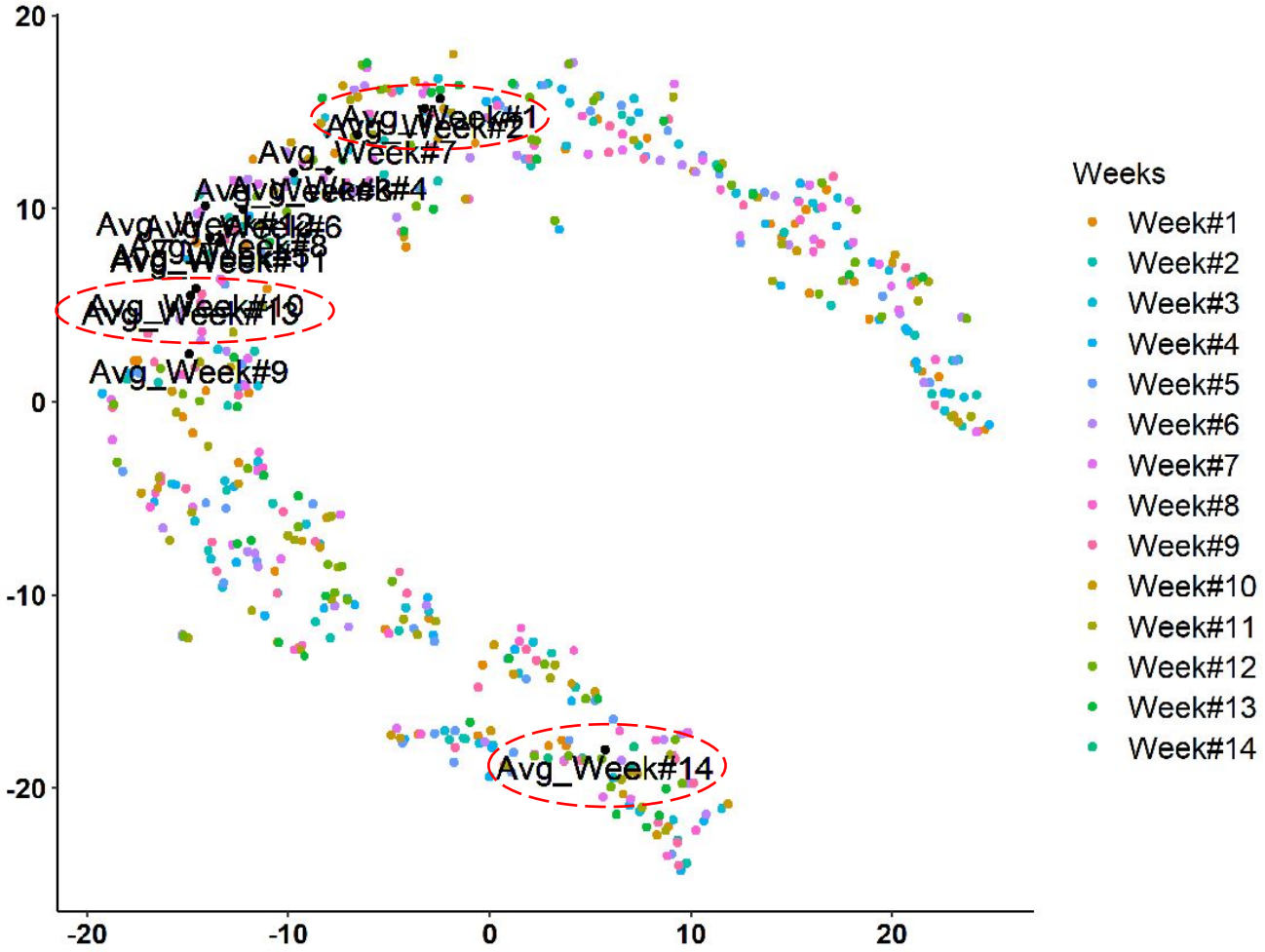
- Course #2 taught to graduate students (Non-industrial projects)
- Course #1 and course #3 taught to undergraduate students (Industrial projects).



Results: characteristics of the student summaries based on text features grouped by teams.



Results: characteristics of the student summaries based on text features grouped by weeks.



Conclusion

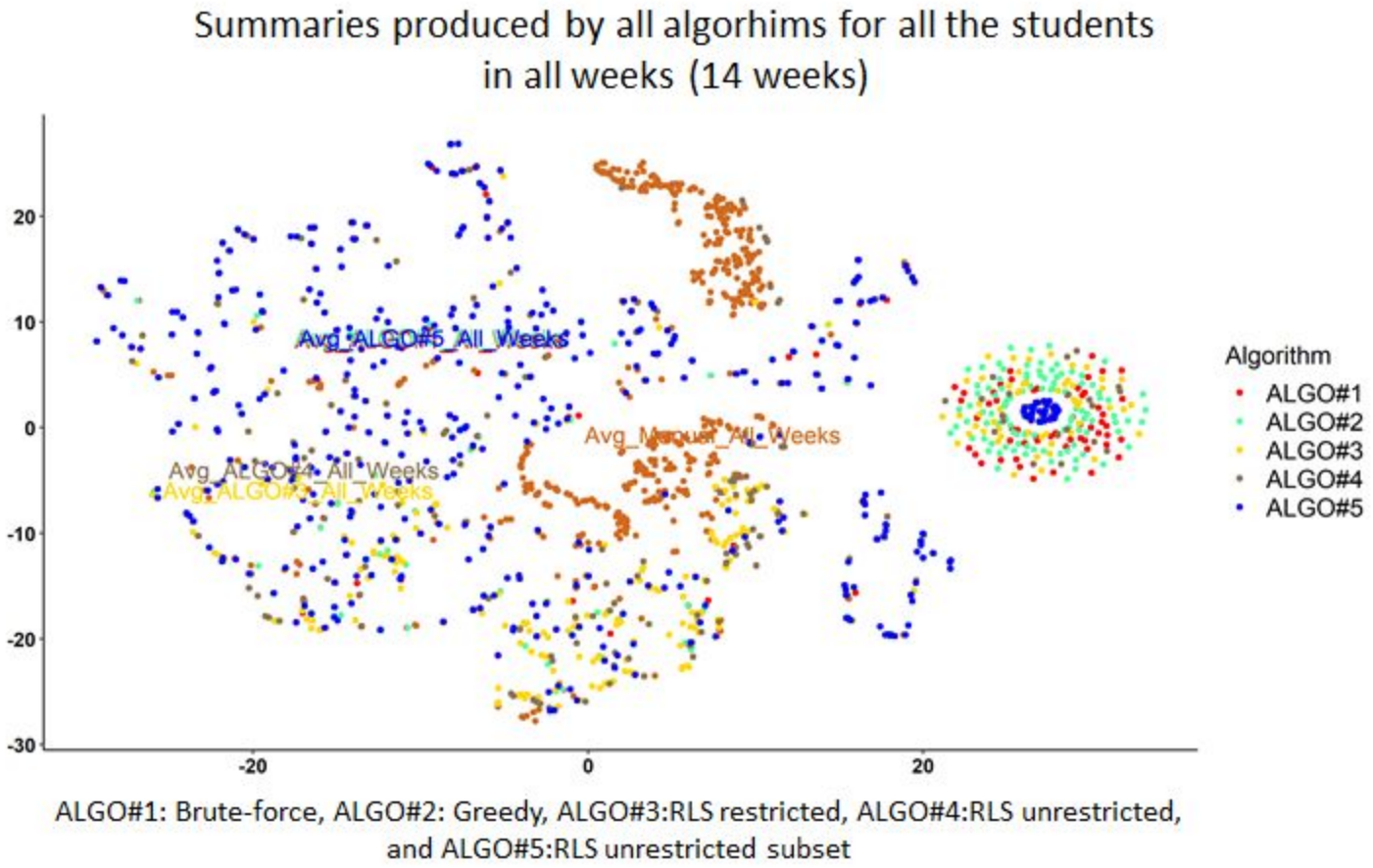
- Our approach of utilising t-SNE to interpret the students' summaries data at different grouping levels using the 27 features allows us to identify summaries that can serve as "gold standard" summaries.

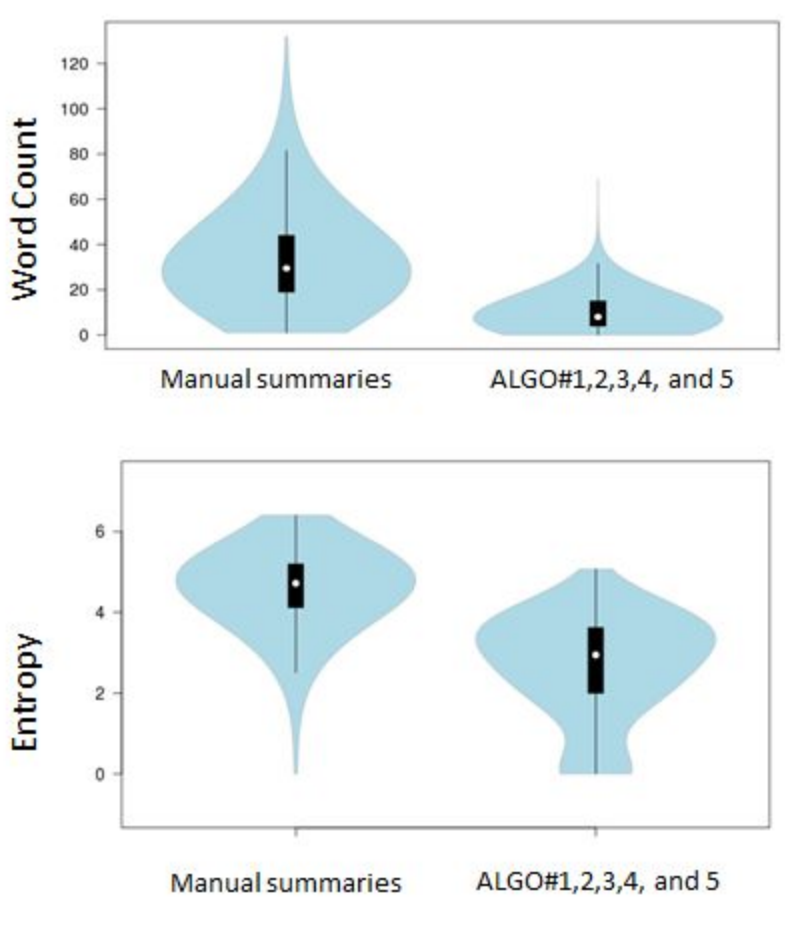
- We will use these to evaluate our future work on extractive summarisation techniques.

Next steps.

- Sentences in software artefacts that are close to the students' summaries.
- Cosine similarity as a similarity measure
  - Texts similarity (bag of words with term frequency)
  - Features similarity (in 27-dimensional space).
- Optimisation methods:
  - Brute-force search
  - Heuristic search
    - Greedy search
    - Local search (restricted, unrestricted, and unrestricted subset)

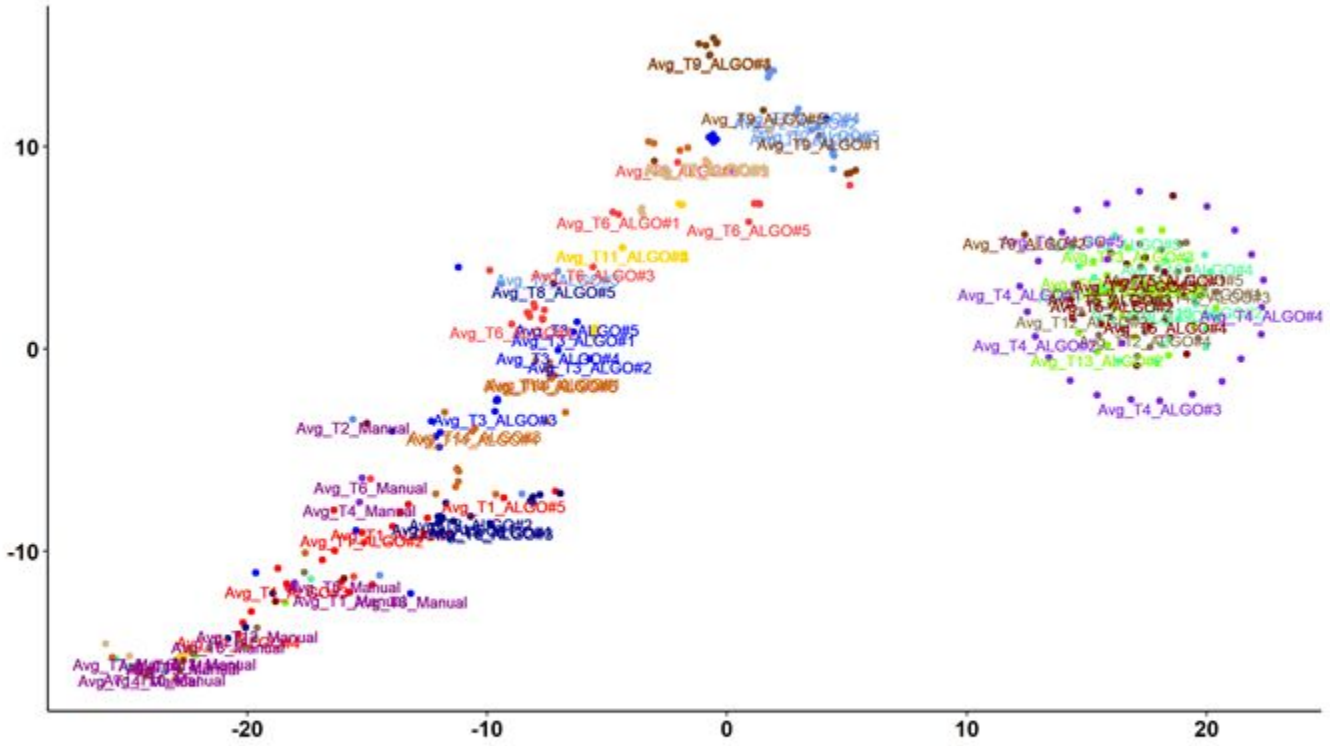
Preliminary Results: Issue title's artefacts.

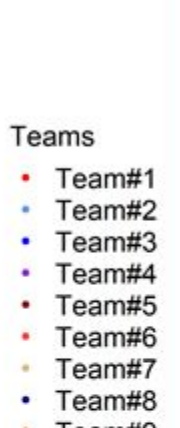




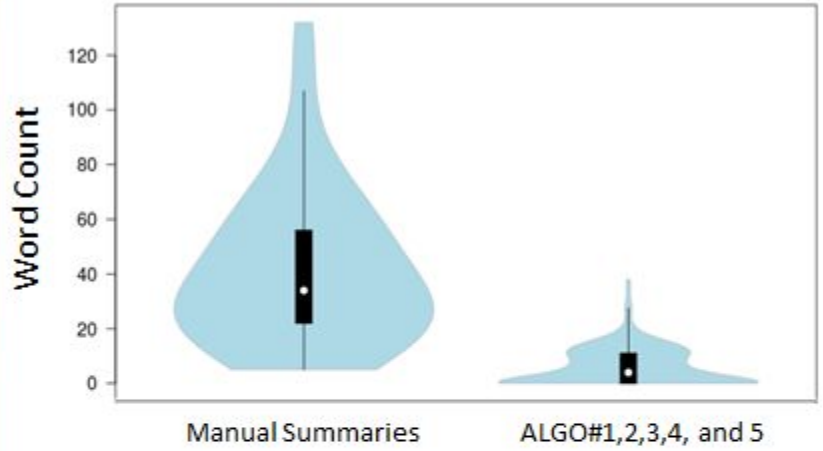
Preliminary Results: Issue title's artefacts

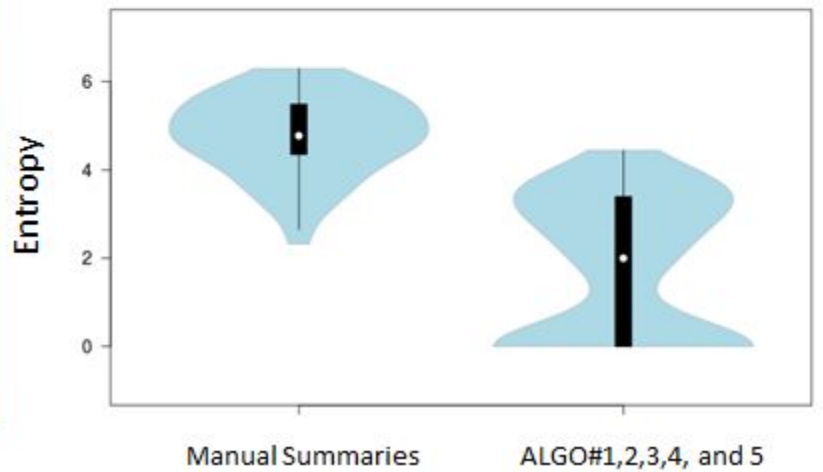
Summaries produced by all algorithms for particular team in weeks 1





ALGO#1: Brute-force, ALGO#2: Greedy, ALGO#3:RLS restricted, ALGO#4:RLS unrestricted, and ALGO#5:RLS unrestricted subset





Next steps.

- There are 15 types of software artefacts we are interested in.

- Measure the similarity between the text in issue title and the students' summaries for a given time window.
  - Students' summaries grouped at different levels (weeks, teams, and courses)

Contact us:
Mahfouth Alghamdi, Christoph Treude, and Markus Wagner
givenname.familyname@adelaide.edu.au