100 Years of Dystopian Novels:

A Computational Analysis of Core Primitives

Melina L. Jander

September 28 2017, Doctoral School on Digital Humanities Symposium

GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Table of Contents

Introduction

About Me

A literary scholar with a focus on contemporary (German) literature.

A member of the eTRAP Early Career Research Group, an interdisciplinary and international team funded by the German Ministry of Education and Research (BMBF) and focussing on Automatic Text Reuse Detection.

Start of my project: April 2017.

Dystopian Novels — 100 Years — Computational Analysis

Dystopia […] is predominantly a modern literary phenomenon of the twentieth century. […] dystopia reverses, mistrusts, and parodies the ideal of a perfectly regulated utopian state, often unintentionally inclined towards totalitarianism. […] dystopia holds up a hellish mirror [to the reader] and describes the worst of all possible futures. […] dystopia presupposes and thrives on the correlation and similarity of the present social order and the near-future scenario. (Mohr, 2005)

Dystopia […] is predominantly a modern literary phenomenon of the twentieth century. (Mohr, 2005)

- "predominantly": the roots of dystopian fiction can be found much earlier, in utopian fiction:

John Stuart Mill coined the word [*dystopia*] in 1868 (Aldridge 1984). Mill had in mind Jeremy Bentham's *cacotopa*—"evil place"—which exactly fits the sense of the definition, but neither term seems to have caught the imagination of critics for the next hundred years. *Dystopia* is preferable to *anti-utopia* for two main reasons: Rhetorically, it exactly reverses the common misreading of More's *eutopia* [Thomas More: *Utopia*, 1516]. (Sisk, 1997)

Dystopian Novels — 100 Years — Computational Analysis

Why?

1. Computational methods have not been used yet to analyse dystopian fiction.

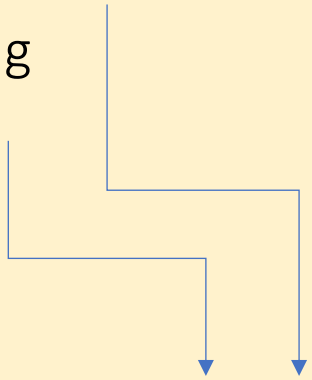
2. With a computational analysis, the corpus for investigation can be much larger.

3. It raises questions of a ‚different' kind: not only the text itself is of interest, but also the computational methods.

How?

— Text Reuse Detection

— Topic Modelling



Topic Detection and Tracking

… but is that enough?

Research Questions

Research Questions

How, if at all, can the methods of text reuse detection and topic modelling be combined in order to localise text reuse and their impact on the different topics present in dystopian novels of the 19th and 20th century?

Which additional computational methods can be used to investigate this corpus of fictional texts?

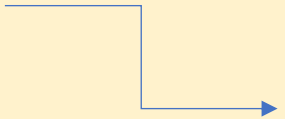
The Corpus

The Corpus: England — America — Germany

Why these three countries?

… popularity, literary history, language skills.

- England: George Orwell's *Nineteen Eighty-Four* (1953)
- America: Ray Bradbury's *Fahrenheit 451* (1949)
- Germany:



No German dystopian novels?

… there are, but not many.

The Corpus: Building It

- Put together manually.
- Based on: literature about science fiction and a list of dystopian book titles available on Wikipedia.
- Data preparation: convert different input formats into consistent output (txt files)

- Choice of novels based on:
  - Popularity
  - Impact
  - Topics (mentioned in summaries)
  - Date and language of first publishing
  - Accessibility

The Corpus: where does the data come from?

Freely available online sources:

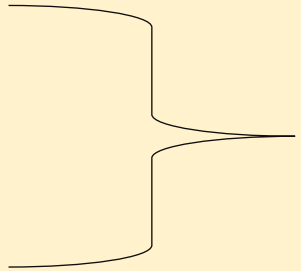
- Project Gutenberg
- Internet Archive
- Deutsches Textarchiv

Formats: epub, txt

Acquisition of eBooks

The Corpus: Basic Numbers

What?	Dystopian novels
When?	1836 – 1977
Languages	American English (n = 38) British English (n = 37) German (n = 16)



Total n = 91

The Corpus: Basic Numbers

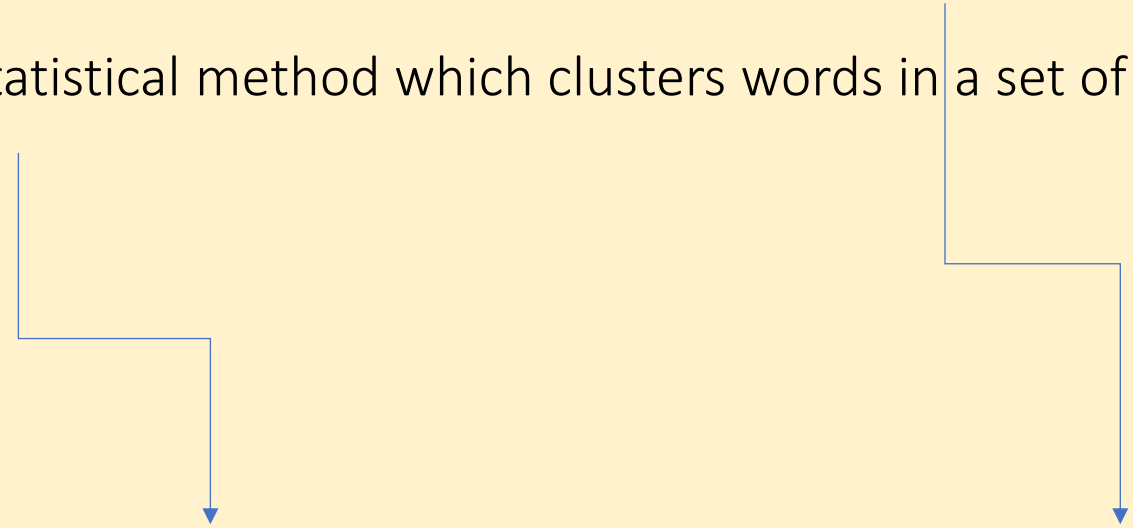
Tokens	American English:	3,167,702	
	British English:	2,615,890	Total: 6,876,439
	German:	1,092,847	
Types	American English:	245,510	
	British English:	190,336	Total: < 492,774
	German:	124,143	

Methodology

Methodology: Text Reuse Detection and Topic Modelling

Text Reuse Detection: spotting written repitition or borrowing of text.

Topic Modelling: statistical method which clusters words in a set of documents.



Topic Detection and Tracking: automatically detecting topically related material and its potential changes in streams of data (e.g., text documents).

Methodology: Topic Detection and Tracking

- TDT can be considered a fruitful method due to its three-fold organisation:
    1. Segmentation
    2. Detection
    3. Tracking

Why can TDT be a consequence of combining TM and TRD?

- TM: questionable output — questionable tool
- TRD: potentially reused entities are too small

Methodology: Topic Modelling

Number of Topics	Topics	Potential Topic Labels
0	garraty mcvries animals stebbins olson baker road farm walk barkovitch napoleon walking crowd back feet passed looked thought pigs animal	**personal names**; farm life; animals; past
1	shevek donald people norman takver chad beninia petra rosalind anarres urras sugaiguntung elihu country bedap shalmaneser michael years physics sabul	**personal names**; people; time; place
2	mark jane adams don't man it's brose dimble director lantano foote nicholas miss leadies i'm wither moment mrs thought studdock	**personal names**; time
3	men women young house people man great lord professor life woman chester dick college work girl love lady beautiful girls	humankind; housing; professions; females
4	arctor barris leacock thought fred carter mrs donna bobby luckman zoo martha house godmanchester bob car englander director people simon	**personal names**; places to live; humankind
5	alvin city diaspar knew earth hilvar great lys world council machine robot khedron long mind time jeserac central ship strange	**personal names**; fantasy world; machines; time (travel)

Table 1: Topic Models created with MALLET (20.09.2017)

Methodology: Topic Detection and Tracking

- TDT can be considered a fruitful method due to its three-fold organisation:
    1. Segmentation
    2. Detection
    3. Tracking

Why can TDT be a consequence of combining TM and TRD?

- TM: questionable output — questionable tool
- TRD: potentially reused entities are too small

Methodology: Text Reuse Detection

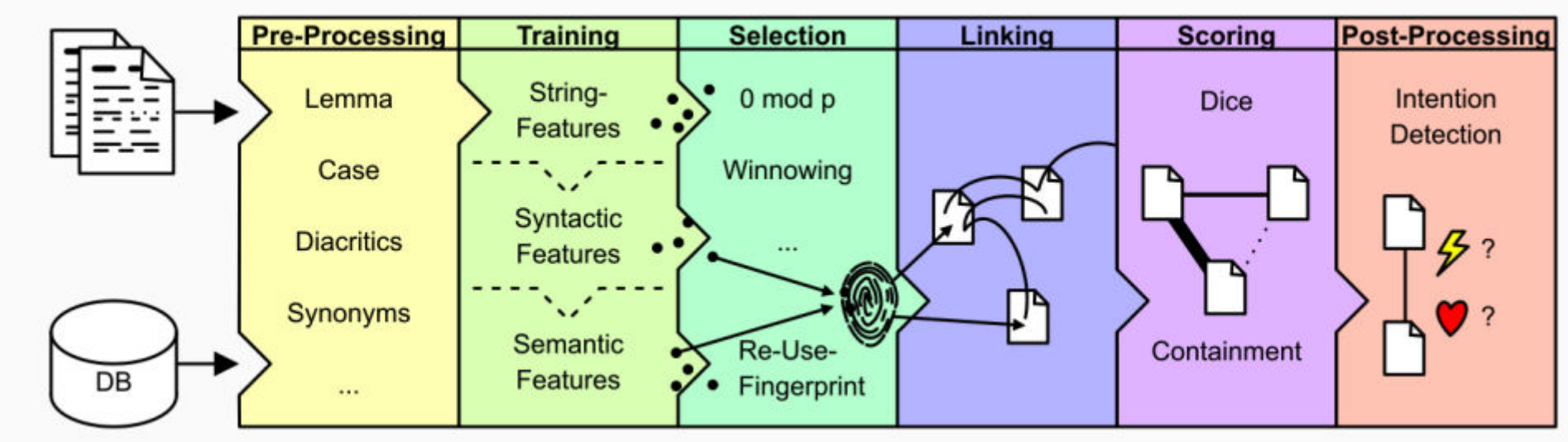


Figure 1: TRACER's detection task in three steps.

TRACER is language-independend. Tested on: Ancient Greek, Arabic, Coptic, English, German, Hebrew, Latin, Tibetian.

Preliminary Findings

COMPUTATIONAL — MANUAL — COMPUTATIONAL

TM and TRD: insufficient for investigating this particular corpus of fiction.

TDT: a potentially useful consequence of TM and TRD.

➢ Challenge: to generate outputs that raise more questions on the literary corpus than on the applicability of the software.

Computational — Manual — Computational

WorldCat: description and categorisation of the novels.

Summaries: detection of topics.

- Difference Analysis: determine discriminatory terms by analysing the different distributions of words in texts.

BabelNet: investigation of the concepts present in the novels.

Computational — Manual — Computational

STYLO: word lists for the English and German corpus.

Voyant: visualisation of the insights gained manually.

- Both the manual investigations and the computational tests show that the topics the dystopian novels should contain are not as present as one could assume.
  - Zipf‘s law.

Next Steps

Combining Computational and Manual Investigations

TM: run more tests with modified stopword lists.

TRD: prepare the data and run TRACER on it.

TDT: see whether it generates results which allow a deeper insight into the corpus.

- Look for alternative approaches: Word Embeddings, Sentiment Analysis?

BabelNet: explore (historical) encyclopaedias and dictionaries to investigate the potential change of meaning of certain terms.

Acknowledgements

I would like to thank Marco Büchler (eTRAP) and Gerhard Lauer (University of Basel) for their inspirational support.

Visit eTRAP:

`http://etrap.eu`

`contact@etrap.eu`