Generating Acrostics via Paraphrasing and Heuristic Search − Final Presentation

Bruno Soares Fillmann
Fernando Bombardelli da Silva
Jürgen Bauer
William Bombardelli da Silva

Technische Universität Berlin
Datenbanksysteme und Informationsmanagement
DBPRO − Database Projects (WS 2014/2015)

09.02.2015

Organization

Goal of the Project

Goal: Implement the methods and techniques to generate acrostics for the German language as described in the paper *Generating Acrostics via Paraphrasing and Heuristic Search* of Benno Stein, Matthias Hagen, and Christof Bräutigam, published in August 2014 [1].

Subtasks of the original goal

- Understand the paper and the method
- Design an architecture for the application
- Collect comparable German text corpora, statistics and libraries (TEX Hyphenation, NetSpeakAPI, Thesaurus, etc.)
- Implement the most promising paraphrasing operators (line break, hyphenation, wrong hyphenation, word insertion/deletion, synonym, spelling)
- Implement the A^*-algorithm
- Evaluate the results

How have we reached the goals?

- UML-diagram for the application
- Sequence diagram for the A^*-algorithm
- Developed the application in Java 8 with Netbeans
- Used various libraries, web services and databases to implement the operators
- Implemented a test application for evaluating the results, with a timeout after 15 min

Which difficulties did occur?

- Synonym operation only inserts synonyms without considering the context (Google N-Gram)
- Lack of powerful grammatical operator
- NetSpeakAPI doesn't yield many results for the German language
- NetSpeakAPI's requests to the web server takes much time
- The empirical choice of costs is not optimal for the search

Which difficulties did occur?

- Organization of the input text as line:
    - More than one letter of the acrostic can be generated at a time, not following the theoretical framework of the A^*-algorithm as described in the paper.
    - **Recommendation:** view the text as one single line and apply line break only to generate a letter. When the acrostic is generated, break the lines according to the constraints.

Unreached Goals

- Only the most promising operators were implemented.
- No real operator to generate the first letter of the acrostic
- No operator **not** mentioned in the paper was coded.
- No local databases for *n-gram* were implemented.
- No sophisticated quality measures were implemented.

Example

Original Text: [1]
Billig nach Hamburg. 24/7 taxidienst bietet ihnen kostengünstige und
zuverlässige verkehr aus Kiel. Mit erfahrung in der branche wir
garantieren die niedrigsten preis nach Hamburg. Gleiche preis gilt
auch wenn sie taxi aus Kiel stadt nach Hamburg nehmen.

Final Text:
B illig nach Hamburg. 24/7 taxidienst bietet ihnen kostengünstige
u nd zuverlässige verkehr aus Kiel. Mit erfahrung in der bran-
c he wir garantieren ihnen die niedrigsten preis nach
H amburg. Gleiche preis gilt auch wenn sie taxi aus Kiel stadt nach
Hamburg nehmen.

---

[1] http://german.airportljubljana.co/Billig_nach_Klagenfurt.php

Results

References

📄 Benno Stein, Matthias Hagen, and Christof Bräutigam. *Generating Acrostics via Paraphrasing and Heuristic Search*.
In Junichi Tsujii and Jan Hajic, editors, 25th International Conference on Computational Linguistics (COLING 14), pages 2018-2029, August 2014. Association for Computational Linguistics.

📄 Spiegel Online. *Griechenland und die Euro-Zone: Der fast unmögliche Rausschmiss*.
http://www.spiegel.de/wirtschaft/soziales/griechenland-euro-austritt-waere-moeglich-aber-kompliziert-a-1011361.html. Am 07. Januar 2015

📄 Hilko. *Blau ist keine traurige Farbe*.
https://deutschlich.wordpress.com/2013/03/12/blau-ist-keine-traurige-farbe. Am 07. Januar 2015

📄 *Redis*. http://redis.io/. Am 07. Januar 2015

📄 *Synonym - OpenThesaurus - Deutscher Thesaurus*.
https://www.openthesaurus.de/. Am 07. Januar 2015

Questions?