Malwina Kotowicz
Department of Computer Science & Media Technology
Linnaeus University
mk224ih@student.lnu.se

Introduction

Thematic Analysis is a method of identifying and analyzing patterns of meanings in a dataset. It illustrates themes that are important in description of the phenomenon under study (Joffe, 2012). As one of the most common methods within the field of qualitative analysis, it's widely used with data from, for example: field research, interviews, observation, solicited diaries and story completion (Braun and Clarke, 2006). The method consists of few phases, one of which is *coding* (tagging items of interest in data with a label: a few words or a short phrase). Such *content analysis* helps to primarily identify *themes* in next phase.

In the final project proposal, I described the design and implementation of an auxiliary tool for thematic analysis. The preliminary purpose of such prototype would be to serve child psychologists and counselors in initial assessment of child welfare. *Behavior Diaries* (solicited diaries) and story completion are among main qualitative research methods and are commonly used in developmental psychology (Gott, 2015; Lämsä et al, 2012). Children are often asked to journal about their everyday life or given topic. The tool I proposed could be deployed as next step in analysis of such text.

With the help of this algorithm, users can facilitate following processes in their thematic analysis research: accessing online data of big volume, extracting *the essence* (information of interest), analyzing the content, plotting the most interesting data pieces in multiple way, training NLP machine learning model on data they choose, performing topic modeling on chosen data (or extraction of data).

As my case study, I've chosen three children's blogs (due to similarity of blogs to journals and diaries) of high originality and distinct leading topics to test the tool on: jakes-bones, lidbemchild, neversconds. My preliminary goal was to present the diversity in children's writing and provide an insight into tendencies, attitudes and dominating moods of what the same age children journal about. The outcome of this algorithm's run is supposed to serve as *an orientation* platform for a researcher in the field of thematic analysis.

Within this case study, the tool will: a) access the content of eight consecutive years (or months) of blogging activity for each blog (libdemchild.com: years 2010-2017, jakes-bones.com: years 2009-2016, neverseconds.com: June 2012-December 2012 and year 2013 due to scarcity of blogging activity in this case), b) scrape the data from each post in chosen period, c) analyze the content of each post in all years and extract the most interesting data, d) use the same data to train simple machine learning model and visualize recurring topics that can be seen in each child writing.

Such dash board of each child blogging activity is sort of a diagnosis of their condition in the environment and society as whole. It can be an important source of information for parents or counselors in the world that praises strong online presence.

The purpose of this report is to provide a description of the achievements, as well as the challenges within the final project design and implementation. This document is therefore sort of a *manual* to the final tool produced.

Approach to solving the given problem and tasks

To run this project, it's necessary to install additional python libraries by:

pip install nltk

pip install genism

pip install pyLDAvis

pip install IPython

The purpose IPython instalation is to allow you visualizations of LDA topic modeling on localhost. However, this step could be skipped if the files *ML_model_blogname* and *LDA-_topic_modeling_blogname* are run jointly in Jupyter Notebook instead of separately in any Python IDE (here: Sublime Text). This option is recommended as it allows training the model first and performing topic modeling next, in the same file, but independently in another cell, which results in shorter runtime, cleaner code and concise file, as well as better access to obtained results. Jupyter Notebook files for topic modeling and ML_model are uploaded to the Moodle in a folder *jupyter notebook files*, so that the user can choose most convenient way to use the project's code.

Outcomes/Analysis of results

The code for described auxiliary tool is organized into five distinct functional files (similar to what was proposed in sections *Major blocks of code* and *Expected outcomes* of final project proposal):

1) *blogname_scrape.py3* for scraping the content of each post in each blog in defined years of blogging activity and saving it to *blogname_scrape_year.csv* file (saved information: date, post's title, post's content)
2) *blogname_read.py3* for sentiment analysis of post's content, converting date to more readable datetime format and saving it to *blogname_year_headline_polarity.py3* (saved information: headline, sentiment's polarity, polarity assessment [neu, pos, neg], date in datetime format)
3) *blogname_dash.py3* for reading content of previously saved files, converting them to dataframes, plotting the data and allowing interactivity by dash *@app.callback* functions
4) *ML_model_blogname.py3* for Word2Vec NLP model training (on real data extracted in previous steps: the results of training on limited amount of data can bring about

interesting observations, e.g.: how well can the model learn just on children's writing?), finding synonyms of words, performing mathematical operations *on words*

5) *LDA_topic_modeling_blogname.py3* for identifying topics in the sets of text in documents (here, low *alpha* α (e.g.; 0.1) allows each document to be composed of only few dominant topics, whereas high *alpha* α (e.g.; 1) allows the opposite. It's recommended to set *alpha* to c.a. 0.5-0.6 (adjustable on the graph).

Data in section 3) is plotted in following way (for each year, each blog):

- Sentiment's polarity of each post is presented in bubble scatterplot (where size=polarity), with post's date, post's title and polarity on the hover, as shown on fig.1 below

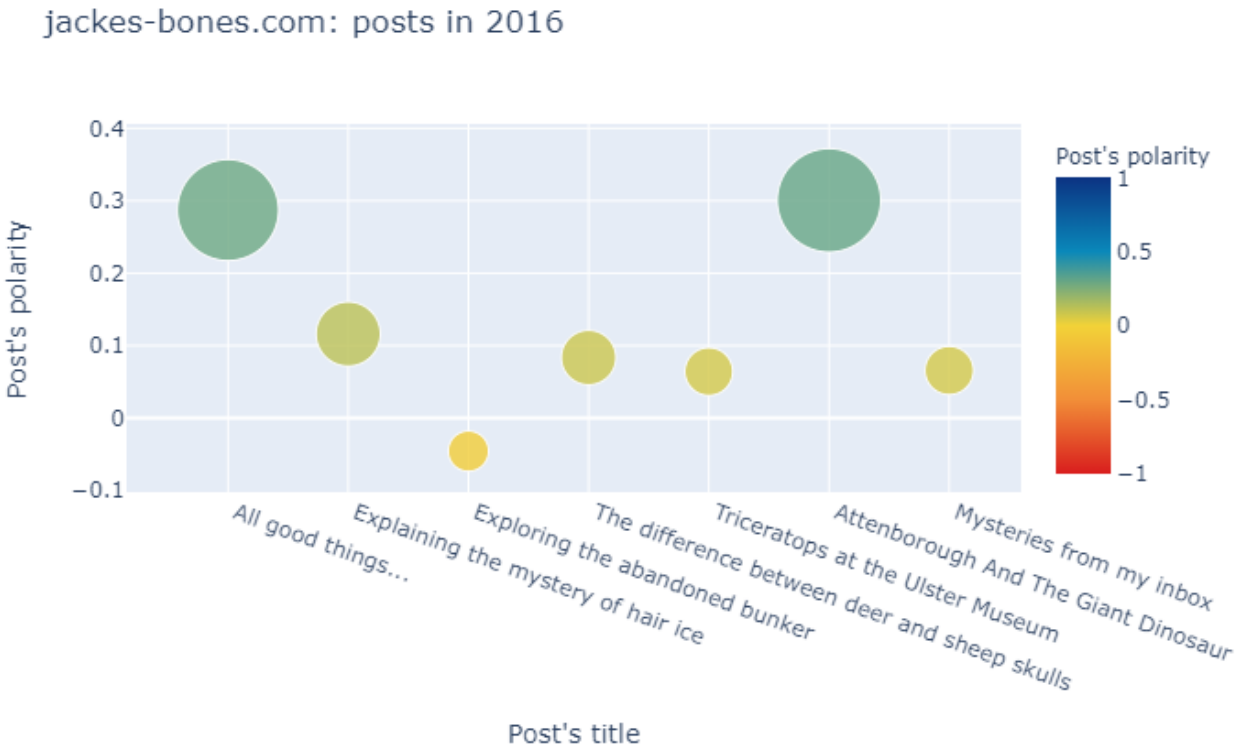

Fig1. Bubble graph of post's polarity in 2016 at jakes-bones.com.

- Bar chart of frequency distribution of 25 most frequent words in a given year with word and its frequency on a hover, as shown on fig.2 below (see next page)

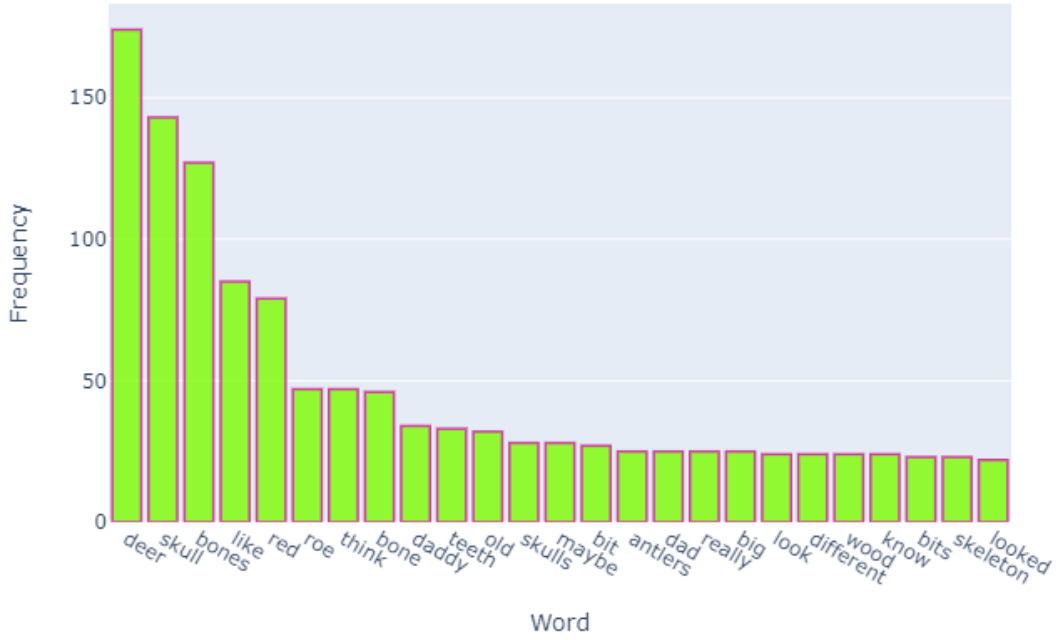

Fig 2. Bar chart of frequency distribution of 25 most frequent words in 2016 at jakes-bones.com

- Wordcloud of frequency distribution of 25 most frequent words in given year, with word and its frequency on hover, as shown on figure 3 below.

Fig. 3 Wordcloud of frequency distribution of 25 most frequent words

It is necessary to mention that plotly does not provide support for wordclouds and such graph type does not exist under dash/plotly. Wordcloud graph's code was written independently to mimic what is understood as *wordcloud*, but has therefore some disadvantages, like the issue of words overlap (solved partially by limiting xaxis length to numbers of words and yaxis value to unique for each word) or discrepancies in word sizes, as the size depends on the frequency (solved partially by normalizing the sizes). As plotly provides zoom in/out

features, these issues are not very bothersome, and the user can see the data as presented below (fig. 4 or fig. 5)

Fig.4 Zoom in of wordcloud in 2012-06 at neverseconds.com

Fig. 5 Adjusting width and height in dash figure (here: width=1500, sizes not normalized) gives a wider overview of frequent words when needed

- Annotated heatmap of the lengths of first 100 sentences in text with sentiment's *diagnosis* (whether it was negative, positive or neutral). On the hover: length and polarity of a sentence, as shown on fig. 6 below (see next page)

Heatmaps can be interpreted as *an author's fingerprint* only when they consist of a sample of text that has been extracted from consecutive chapters or sections that logically flow in text (instead of random sampling within the text). Sentence length is an indicator of style that can be used to estimate how good the rhythm is and how it can be preserved in translation. It has been shown that distribution of sentence length (as in heatmaps) can be an important factor in authorship attribution as more reliable marker than the average sentences length. (Keim D., Oelke D.)

For the simplicity of dashboard, annotated heatmap was created with plotly, although it stated otherwise in final_project_proposal (Seaborn). Therefore, similar heatmap was created with Seaborn and Matplotlib, implementing the code:

import matplotlib.pyplot as plt
import seaborn as sns
fig, ax = plt.subplots(figsize=(12, 7))
sns.heatmap(results, annot=labels, fmt='', cmap='RdYlGn', linewidths=0.3, ax=ax)

Seaborn heatmap is shown on fig. 7 below.

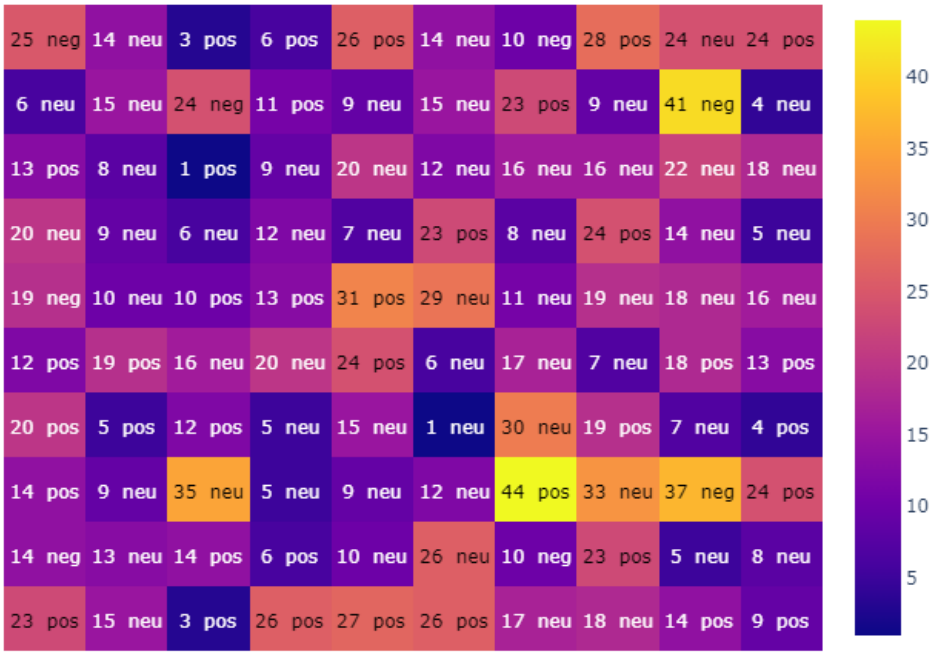

Fig. 6 Annotated heatmap of the lengths of first 100 sentences in text with sentiment's *diagnosis* (plotly)

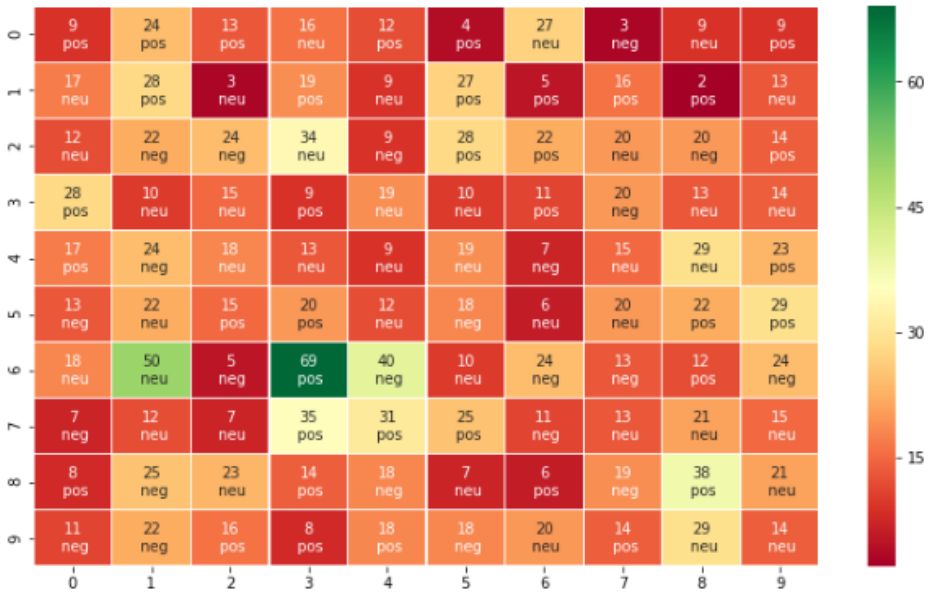

Fig. 7 Annotated heatmap of the lengths of first 100 sentences in text with sentiment's *diagnosis* (Seaborn)

- Scatterplots (**subplots**, line charts) for all sentiment's polarity *diagnosis* (negative, positive, neutral) with dates of sentiment *dropdown* (e.g.; from positive to negative) in all years of blogging activity to provide thorough overview and comparison (this graph is the same for each year, it does not update with *@app.callback*), see: graph in dashboard panel.
- Similar to previous one, scatterplot (**in one plot**, line charts) for all sentiment's polarity *diagnosis* (negative, positive, neutral) with dates of sentiment *dropdown* (e.g.; from positive to negative) in all years of blogging activity to provide thorough overview and comparison (this graph is the same for each year, it does not update with *@app.callback*) as shown below:

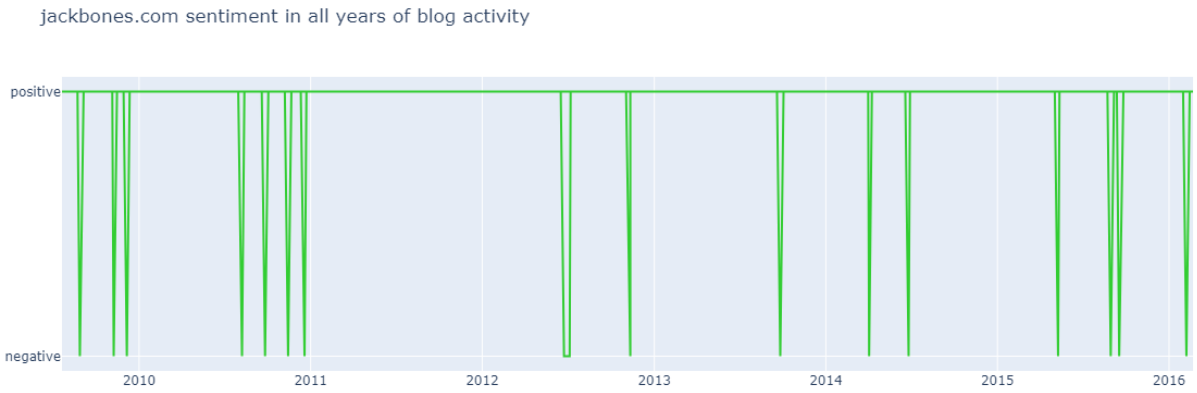

Fig. 8. Scatterplot of all sentiment's polarity in all years of blog activity at jakes-bones.com

Important note: At the very first run, it is possible to get callback error while running dash app on localhost (error updating a figure, particularly freq-dist or wordcloud). It is enough to reload the website for the plots to appear in the right place.

It is also worth mentioning that once run dash board needs to be hold on by accessing **task manager>>processes>>python>>**stop(all)processes if data from another blog is supposed to be visualized next. This way allows localhost to reload.

The outcomes of Machine Learning model training and LDA topic modeling

1) ML model training allows searching for synonyms in text content and mathematical operations (as mentioned before). The results obtained in this step depends on the *volume* of data used to train the model (over 13000 sentences in jakes-bones.com versus over 3000 sentences in libdemchild.com). This parameter may decrease the accuracy of the model in searching for synonyms and therefore:

Synonyms of word "good" in jakes-bones.com:
[('great', 0.9999316334724426),
('wildlife', 0.9999279379844666),
('amazing', 0.9999215006828308),
('nice', 0.9999212622642517) …]

Synonyms of word “baby” in jakes-bones.com:
[('adult', 0.9998760223388672),
 ('young', 0.9998599290847778),
 ('stag', 0.9998427629470825),
 ('female', 0.9998358488082886),
 ('sheep', 0.9998347759246826) …]

“amazing”+”good”–“bad” in jakes-bones.com:
[('read', 0.9997396469116211),
 ('great', 0.9997318983078003),
 ('cool', 0.9997299313545227),
 ('brilliant', 0.9997177720069885),
 ('nice', 0.9997155070304871) …]

Synonyms of word “good” in libdemchild.com:
[('children', 0.4087221026420593),
 ('people', 0.3898881673812866),
 ('child', 0.38699424266815186),
 ('government', 0.3806324899196625) …]

Differences in performances of models (trained on different volume of data) are noticeable (with bigger training sets giving satisfying results in jakes-bones.com).

As an interesting mention, it is recommended to *play around with* different training parameters, like *window* (if the purpose is to get synonyms, the window should be set to 1, maximum 2, if the user aims at finding general context: attributes and relationships between neighboring words, it’s enough to set window to 5-10. With *window=5-10 most_similar()* words found by model can be surprising: e.g.: winter and summer (both being seasons).

Vector size (*size)* can be set to 100, 200, 300, *min_count* if set to 1, the model will consider tokens that appear in text at least once. It is not useful when user deals with content of big rarity, but it can be readjusted to more than 1 if user deals with texts of big diversity (like Wikipedia articles) and it can be assumed that if a token occurs just once in text, it is probably a typo.

2) LDA topic visualization with adjustable *alpha* α provides an overview of topics that appear in total blog content (in given years), as shown on fig. 9

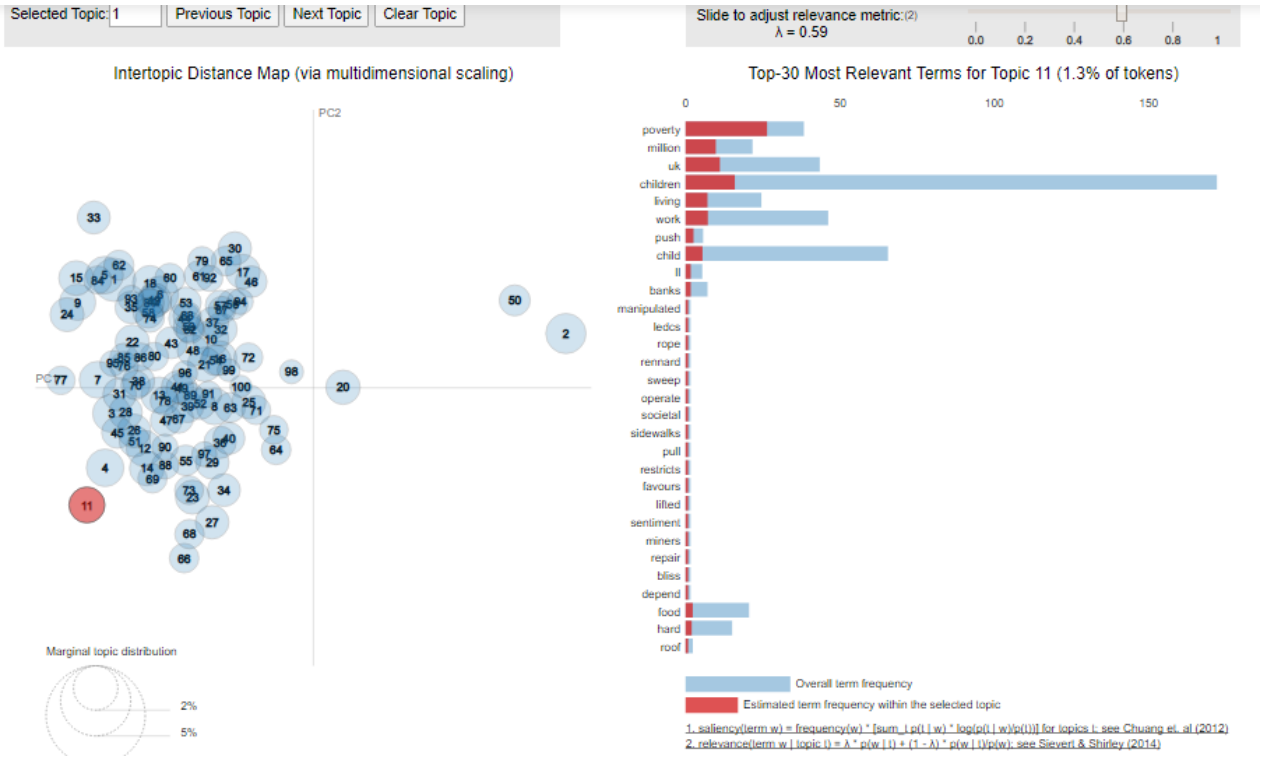

Fig. 9 LDA topic modeling for libdemchild.com, showing topic 11 with tokens: [poverty, million, children, UK, living, work (..)]

Here, LDA topic modeling is a subsidiary tool for illustrating moods and tendencies in children's writing and can help spotting *dark regions* on child *mental map.*

Conclusions and Reflections

Despite the complexity of the task and quite ambitious objectives assumed in final project proposal, this tool for thematic analysis was developed successfully and accordingly to all preliminary assumptions. It demanded a huge amount of time and energy devoted, but the benefits and the amount of new knowledge acquired after all are priceless.

The video presentation of working application can be found here:
https://youtu.be/XXiXq8odXgM

This code will be soon uploaded to github where it can get new opensource life and be developed by the community.

REFERENCES

1. Joffe H. Thematic analysis. In: Harper D, Thompson AR, editors. Qualitative research methods in mental health and psychotherapy. Chichester: John Wiley & Sons, Ltd.; 2012. p. 209–223.

2. Keim D, Oelke D. Literature Fingerprinting: A New Method for Visual Literary Analysis, Visual Analytics Science and Technology, 2007

3. Sue Gott (2015). Behaviour Diaries: An Assessment Tool for Supporting Children with Behavioural Difficulties, Speechmark Publishing

4. Tiina Lämsä, Anna Rönkä, Pirjo-Liisa Poikonen & Kaisa Malinen (2012) The child diary as a research tool, Early Child Development and Care, 182:3-4, 469-486

5. Virginia Braun & Victoria Clarke (2006) Using thematic analysis in psychology, Qualitative Research in Psychology, 3:2, 77-101

Web-based material, Linnaeus University and others:

Blogs:

1. Libdem child: http://libdemchild.blogspot.com/
2. Jake's Bones: http://www.jakes-bones.com/
3. NeverSeconds: http://neverseconds.blogspot.com/

Other resources:

1. Word2Vec: https://medium.com/explore-artificial-intelligence/word2vec-a-baby-step-in-deep-learning-but-a-giant-leap-towards-natural-language-processing-40fe4e8602ba
2. LDA topic modelling: https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158
3. DataWorkshop tutorials for ML challenge participants: https://dataworkshop.eu/en/
4. Sentiment analysis tutorials: https://towardsdatascience.com/real-time-twitter-sentiment-analysis-for-brand-improvement-and-topic-tracking-chapter-2-3-1caf05346721