LAPPEENRANTA-LAHTI UNIVERSITY OF TECHNOLOGY LUT
School of Engineering Science
Software Engineering

PALAUTE: AN ONLINE TOOL FOR TEXT MINING COURSE FEEDBACK USING TOPIC MODELING AND EMOTION ANALYSIS

Examiners: Assistant Professor Antti Knutas
Professor Jari Porras

TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT
School of Engineering Science
Tietotekniikan koulutusohjelma
Niku Grönberg

Palaute: Verkkotyökalu kurssipalautteen tekstilouhimiseen hyödyntäen aihemallinnusta ja tunneanalyysia

Diplomityö 2020
78 sivua, 19 kuvaa, 5 taulukkoa
Työn tarkastajat: Apulaisprofessori Antti Knutas
Professori Jari Porras
Hakusanat: tekstinlouhinta, opiskelijapalaute, rakenteellinen aihemallinnus, tunneanalyysi
Keywords: text mining, student evaluation of teaching, structural topic model, emotion analysis

Opiskelijapalaute on todettu hyödylliseksi metodiksi opetuksen kehittämisessä. Palaute nojaa yleensä Likert-tyyppisiin kysymyksiin niiden helpon analysoinnin takia, ja jos avoimia kysymyksiä käytetään, niin niiden analysointi jää yleensä vain vastausten läpi lukemiseen. Tämä rajoittaa avointen kysymysten hyödyllisyyttä, vaikka ne ovat vähemmän rajoittavia kuin Likert-tyyppiset kysymykset ja sallivat tarkemman palautteen antamisen. Palaute (Plot, analyze, learn and understand topic emotions) on työkalu, joka kehitettiin kurssipalautteen avoimien kysymysten analysoimiseen. Tavoitteena oli tehdä datan ymmärtämisestä helpompaa. Palautteessa yhdistetään aihemallinnusta ja tunneanalyysia datan kiteyttämiseen. Asiantuntijoiden arviot ja demo osoittavat, että työkalusta on hyötyä opiskelijapalautteen analysoinnissa. Tämän lisäksi tutkittiin suomenkielisen tunnesanaston typistämisen vaikutusta tunneanalyysiin. Tulokset kuitenkin näyttivät alkuperäisen sanaston suoriutuvan paremmin kuin typistetyn sanaston.

ABSTRACT

Lappeenranta-Lahti University of Technology
School of Engineering Science
Software Engineering
Niku Grönberg

Palaute: An online tool for text mining course feedback using topic modeling and emotion analysis

Master's Thesis 2020
78 pages, 19 figures, 5 tables

Examiners: Assistant Professor Antti Knutas
Professor Jari Porras
Keywords: text mining, student evaluation of teaching, structural topic model, emotion analysis

Student evaluation of teaching has been accepted as a useful method for improving teaching. The evaluation usually relies on Likert-type questions as they are easier to process, and if open questions are used, they are usually not analyzed beyond reading through them. This limits the usefulness of open questions, even though it is evident that they allow for more accurate feedback from the students, and they are not as limiting as Likert-type questions. Palaute (plot, analyze, learn and understand topic emotions) was created as a tool for addressing the answers to open questions in student course evaluation surveys with the goal of making understanding the data easier. Palaute combines topic modeling with sentiment and emotion analysis to summarize and create insights from the data. Expert reviews and demonstration show that the tool is useful in its intended task. Additionally, the effects of stemming a Finnish emotion lexicon were investigated to improve the emotion analysis performance, with results leaning towards the original lexicon.

ACKNOWLEDGEMENTS

Simo inspired me to get those gains.

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

ABSA	Aspect based sentiment analysis
CSV	Comma-separated values
CTM	Correlated topic model
DMR	Dirichlet-multinomial regression
DPM	DepecheMood
DTM	Document-term matrix
ESN	EmoSenticNet
GPLv3	GNU general public license v3.0
IS	Information system
ISM	Information seeking mantra
LDA	Latent Dirichlet Allocation
LSTM	Long short-term memory
MPQA	Multi-perspective question answering
NLP	Natural language processing
NRC	National research council Canada
RNN	Recurrent neural network
RQ	Research question
SAGE	Sparse additive generative
STM	Structural Topic Model
SVM	Support vector machine
SO-CAL	Semantic orientation calculator
TDPM	Topic based DepecheMood
tf-idf	Term-frequency inverse-document-frequency
t-SNE	t-Distributed stochastic neighbor embedding

1 INTRODUCTION

It has been almost a century long discussion about whether student evaluations of university courses are useful or not, some claiming students do not have the academic training to understand the pedagogic requirements for teaching a course, while others claim student evaluation is paramount for improving courses since they have the perspective of receiving the education (Jordan, 2011). Marsh argues that most of the fears about student evaluations are based on two poorly conducted studies and student evaluations are overall multidimensional, reliable, relatively unbiased and pose a utility for the teaching staff (Marsh, 1984). Although, student evaluation should not be used as a measurement of teaching performance, and instead should only be used as feedback to improve the courses (Marsh, 1984; Zabaleta, 2007). Overall, student course evaluations are widely adopted and commonly used as a way to improve the courses and the level of teaching, although the benefits of using student course evaluations are tied to the amount of effort used implementing the suggestions made by students (Kember et al., 2002).

LUT University collects student evaluations through a non-mandatory anonymous online survey after each course. This practice was started in 2004. The student union arranges sending the surveys and collecting the responses, after which they are handed over to the university staff. The surveys are emailed to the students that partook in the courses after the courses have finished. Answering these voluntarily feedback forms is incentivized with improving the courses and usually some small gift cards are raffled among the respondents. The course teachers are then required by the university to go through this course feedback and post a response to the course participants with the main themes that came up from the feedback and what changes are going to be implemented in the course going forward.

Going through the course feedback can be a daunting task, especially on the freshman courses, some of which are mandatory for every student in the university. These courses can have over 500 participants, so even if only half of the students give feedback, it is still a laborious task for the course teacher. Most of the courses, of course, have much lower numbers of participants.

The goal of collecting student evaluations of the courses should be to improve the courses, as other use cases like personnel comparisons and measuring teaching performance should not be based solely from student evaluations (Marsh, 1984; Zabaleta, 2007). As the students are the ones receiving the education, they should have valuable information about the positive aspects of the course as well as the issues they faced because of the course design. Depending on the survey instrument, answers received from the students give insight for example in learning during the course, enthusiasm of the lecturer, group work, examinations and the level of workload (Marsh, 1984). The use of qualitative questions allows the students to give suggestions, observations and frustrations, and these can be specific to issues not covered by quantitative questionnaires (Jordan, 2011). When the same suggestion or observation is offered by multiple students, it can serve as a pointer to a problem (Gottipati et al., 2018).

The problem is systematically addressing the qualitative results, as it is a demanding task usually with no formal guidelines. Thus, the qualitative data is not used as effectively as it could be, and the use of the qualitative data is usually limited to the course teacher (Jordan, 2011). There is a lot of information in the qualitative data that could be used in a larger scope that is not limited to just serving as suggestions for the course teacher. For example, comparing feedback from similar courses to gain information about what works and what does not, based on the course context. This kind of information is hard to induce from quantitative data, as it cannot answer why students liked or disliked the course.

Text mining is a technique that enables analyzing unstructured text with the goal of finding information that is not clearly visible from the data (Garg and Heena, 2011). More precisely, text mining allows, for example, identifying topics shared between multiple documents (Blei et al., 2003; Roberts et al., 2013) and understanding the sentiments or emotions indicated in the text (Hu et al., 2018; Kumar et al., 2019). Text mining tools can be used to systematically analyze the answers to qualitative questions of a survey, somewhat solving the issue of having to analyze the qualitative data by hand.

Multiple different text mining techniques have been demonstrated on course evaluation surveys by, for example, (Koufakou et al., 2016; Sliusarenko et al., 2013). A tool was also proposed for extracting suggestions from the evaluation surveys by (Gottipati et al., 2018). Therefore, it does seem possible and reasonable to apply text mining techniques to student course evaluation surveys.

Topic modeling algorithms are used to extract topics from collections of documents. Applying them to course evaluation surveys would allow for summarizing the course feedback efficiently. Understanding the main points extracted from all the survey responses should be very useful.

In addition to summarizing the main topics, the emotions found in the text can also be analyzed and summarized. Emotion analysis is a text mining technique for extracting the emotions from the text based on the individual words and structures of the text (Kumar et al., 2019). Understanding the emotions of the students answering the survey can yield useful information for improving the course as well as understanding whether the feedback is overall positive or negative.

1.1 Goals and delimitations

The goal of this thesis is to create and evaluate an artefact that is a tool for analyzing the answers to open questions from student course feedback surveys. Creation of this tool follows the design science principles.

The tool should be able to extract useful information from the student's answers that is hard to interpret from the data by hand. For example, understanding the emotions of the respondents can be felt when reading through the answers, but it will be hard to quantify how much of each emotion is in the answers and what it is directed at, especially since we tend feel negative emotions more strongly. So, understanding the emotions from the data should be useful.

The tool should summarize the data in a way that makes understanding the data an easier task than reading through it all. The main structure and main points of the data should be made visible and communicated to the user in a way that is easier than doing it by hand.

The tool should be able to analyze LUT university's student course evaluation answers, as it will be aimed at achieving that. This means that there might be limitations with other kinds of data. There is the limitation of only supporting Finnish and English as they are the languages used in LUT university.

Understanding whether the artefact is useful or not in the context of student course evaluations is the core of this study, as is understanding what kind of information can be learned with the tool from the course evaluations. Thus, the research questions (RQ) are based around evaluating the artefact rather than improving individual courses. The formal research questions of this study are:

RQ1 Can the tool be used to analyze the intended data in a meaningful way?

RQ2 Does the intended user group deem the artefact useful?

RQ3 Can the tool accurately identify emotions from the data?

1.2 Structure of the thesis

The thesis begins by introducing the problem and giving background in the chapter 1. Literature and relevant studies are presented in chapter 2 as to give more background for this study. The research method is specified in the chapter 3. Artefact design is shown in chapter 4, followed by the implementation details of the artefact in chapter 5. The artefact is evaluated in chapter 6 and the evaluation results are discussed in chapter 7. Lastly, the main takeaways from this thesis are summarized in chapter 8.

2 RELATED WORK

This literature review covers the background and work related to text mining starting from student evaluation of teaching. Literature about text mining and more specifically about topic modeling and sentiment analysis is reviewed and some text mining solutions are listed as examples. Lastly, visualization is researched to communicate the text mining results to the user effectively. The literature review is synthesized in Table 1.

2.1 Natural language processing

Natural language processing (NLP) refers to computationally processing speech or written text to gain something useful. Language has evolved into very complex structures capable of conveying ideas and emotions, and this richness while easy for humans to understand, makes it difficult to process it with computers (Stojanovski et al., 2018). NLP deals with three major problems: understanding individual words and their meanings, understanding sentences and their meanings and understanding the overall environment or context. NLP problems are based around for example machine translation, speech recognition and summarizing collections of text. (Chowdhury, 2003)

2.2 Text mining and analysis

There are multiple approaches and goals in different text mining applications and algorithms, but overall the structure of text mining process usually follows the six steps shown in Figure 1 by (Hashimi et al., 2015). The goal of text mining is to extract unknown information from unstructured text data. Text mining is similar to data mining with the exception that data mining deals with structured data, whereas text mining tools are designed to work without structures in the data. Although, it would be wrong to say that text does not have an inherent structure, it is just too complicated to be modeled accurately, rendering it unstructured for data mining applications. (Sanchez et al., 2008)

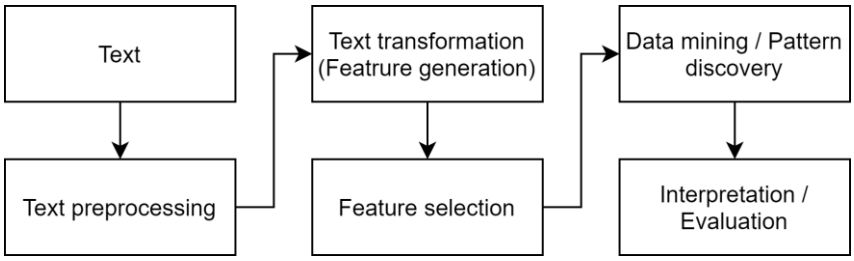

Figure 1. Text mining process

2.2.1 Different types of data in text mining

The first step in text mining is the input to the process. This can be individual documents, or collections of documents of varying sizes. Text mining has been used in literature, for example, with emails, law texts, scientific literature, tweets, online reviews and course evaluation surveys.

Ahonen et al. used text mining techniques on Finnish law texts. Overall the corpus consisted of 759 separate documents. They used episode rule techniques to find out useful information in the text. In practice this means that they mined for frequent phrases and co-occurring words. (Ahonen et al., 1997)

Mohammad and Yang did sentiment analysis on emails, categorized more specifically to love letters, hate mail and suicide notes. The overall corpus sizes were 348 for love letters, 279 for hate mail and 21 for suicide notes. They found that men send and receive emails with more words relating to trust and fear, while women send and receive emails with more words indicating joy and sadness. (Mohammad and Yang, 2013)

Twitter has been a source for multiple studies utilizing text mining. This is likely due to the Twitter API that allows retrieving tweets from the site for example with a certain hashtag. Tweets differ from other documents since they are short (maximum of 240 characters) and they are noisy, meaning they contain emojis, urls, hashtags, slang and typing errors. Text mining Twitter has been used to analyze how other countries view United States as a nation (Lucas et al., 2015). Curiskis et al. tested the suitability of different topical models and clustering algorithms in analyzing tweets. They found that a trained neural network called

word2vec worked best with k-means clustering, since topical modeling suffered from the noisiness of the tweets (Curiskis et al., 2020). Text data from Twitter has also been used in identifying issues students have with studying engineering (Chen et al., 2014).

Different kinds of feedback have been analyzed, for example, online reviews and course evaluation surveys. 27000 hotel reviews were analyzed from the site TripAdvisor.com to find out the main reasons for negative feedback (Hu et al., 2019). Wang and Goh used a total of 9333 game reviews from Amazon to understand what aspects of the games receive more positive feedback and what are the main causes of criticism (Wang and Goh, 2020).

Educational data mining has been important field of research even as early as 1995 (Romero and Ventura, 2007). Multiple different aspects of student evaluations of teaching have been studied, for example classification of Likert-type questions (Agaoglu, 2016), understanding the relations between the teacher's characteristics and their performance (Zhang et al., 2017) and multiple studies about understanding the student behavior in online courses (Romero and Ventura, 2007). Sentiment analysis on responses to qualitative open questions in student course evaluations has been done by (Ahmad et al., 2019; de Paula Santos et al., 2016; Koufakou et al., 2016; Pong-Inwong and Kaewmak, 2016). Qualitative course feedback survey questions have also been text mined for inappropriate comments (Tucker, 2014), assisting in the selection of outstanding faculty (Tseng et al., 2018), extracting the student suggestions (Gottipati et al., 2018) and overall exploratory data analysis using the Leximancer tool (Stupans et al., 2016). Sliusarenko et al. extracted key phrases from course evaluation surveys' open questions and compared them to the quantitative Likert-type questions of the same survey at the Technical University of Denmark. They found multiple different topics from the open feedback and that the quantitative answers match with the qualitative answers only partly (Sliusarenko et al., 2013). Jordan evaluated text mining techniques on course evaluation surveys and found among other results that text mining can be used to extract new information from documents, and while not being as good as manual interpretation of the documents, text mining is close to the human level (Jordan, 2011).

One more example of different application of text mining is the mining of medical literature. There are multiple studies of this but, for example, Feldman et al. used text mining techniques to summarize relations between genes and diseases. In medical research text mining is important, as the corpus of documents grows extremely fast and the corpus already contains millions of documents (Feldman et al., 2003).

2.2.2 Text preprocessing

The second step of text mining is preprocessing the text. This is done to process the text into a machine-readable state. There are multiple steps in preprocessing text and depending on the case, not all of them are necessary. The main text preprocessing steps are stop word removal, stemming and lemmatization. Depending on the language, translation, dealing with compound words, and segmentation might also be necessary. (Lucas et al., 2015)

The first step is turning all the documents into the same format. Collecting documents from different locations might mean that they are in different encodings, in other words, the file type on the computer might differ. Depending on the text mining tool used, the files should be changed to match the required encoding. (Lucas et al., 2015)

The individual documents are usually turned into bag-of-words vectors containing all the unique words and their respective counts in the document. Transforming the text into structured vectors references the text transformation step in Figure 1. Most of the preprocessing steps can be done to either unstructured text or the structured vectors, with some changes in the outcome with for example translation. Depending on the goal, it is possible to first apply preprocessing steps and then turn the documents into vectors or apply preprocessing to the document vectors. (Lucas et al., 2015)

Documents can be turned into structured vectors from unstructured text and corpora can be turned into structured document-term matrixes (DTM). DTM is used to store all the unique words of a corpus of documents and the word counts respective to these documents. DTM takes the document vectors from a document level to the corpus level by containing multiple document vectors, one for each document.

Stop word removal is the process of removing common words that have no meaning in the context of the desired results (Lucas et al., 2015). Common stop words in the English language are, for example, “or”, “the” and “is”. Removing stop words is completely language dependent, and it can affect the results (Fokkens et al., 2013), or in some cases it doesn’t affect the results (Biggers, 2012). It is still accepted that stop word removal should be done at least in topic modeling, as the common removed words, such as articles in English, bear no meaning to any particular topic that exists in the corpus. The application of text mining should be the determining factor for the selection of stop words to be removed, as the goal defines the words that are not necessary (Lucas et al., 2015). One common method is also removing the rarest words from the corpus, as their relevance is low due to the low word count compared to other words (Eler et al., 2018).

Stemming removes the endings of inflected words and leaves just the part that is same in all the inflected forms. Since the relevance of a word to a specific topic is usually same and does not depend on whether the word was in singular form or plural, it does not make sense differentiate between for example “car” and “cars”. For verbs, this means removing the tense, for example “decline, “declined” and “declining” become “declin”. Stemming does not solve the issue of “decline” having multiple meanings (refusing an offer, a value decreasing) depending on the context, although with English the impact of stemming not capturing all the meanings correctly is actually small. (Lucas et al., 2015)

Stemming is an approximation of a more general goal of lemmatization, which means understanding the basic form of a word and grouping the basic forms together. Lemmatization thus requires differentiating between different meanings of a word depending on the context it is used in. (Lucas et al., 2015).

In the case of English, stemming is a great at approximation of lemmatization and the results are almost as good as with lemmatization (Lucas et al., 2015). In the case of Finnish, which is highly inflectional and agglutinative language, lemmatization yields better results in clustering applications of Finnish text than stemming does (Korenius et al., 2004).

Compound words pose an issue, since the meaning of the word as a whole can be different than the meanings of the individual words, but the individual words might still be relevant for topic modeling or clustering (Lucas et al., 2015). Usually the compound words are similar in meaning whether their components appear compounded or separate, but for clustering Finnish text it would seem best to have compound words both compounded and as separated words (Korenius et al., 2004).

Translating text becomes necessary when the corpus is multilingual since text mining different languages would mean the same word in each language is treated as a unique word. While human translation is the best option in terms of quality, for a larger corpus it quickly becomes impossible, making machine translation the only option. Translation can be done to every document or just to the DTM. Translating the whole documents has the advantage of including context in the translation process, but this comes at a cost of having to translate multiple times more characters than if only the DTM is translated. Translating just the DTM brings the issue of not having context for the words, meaning a word can be translated to the wrong meaning if there are multiple possible translations for that word. The results of text mining are dependent on whether the whole documents were translated or just the DTMs. There is also the issue of what language should the texts be translated to. In case of two languages translating the first language to the second language would mean one language is accurate and the other is as good as machine translation can be. The other solution is to translate both of the languages to a third language, so both of the corpora have similar levels of translation error. (Lucas et al., 2015)

2.2.3 Data mining using topic modeling

Topic modeling algorithms, like latent Dirichlet allocation and structural topic model, achieve the two steps after text transformation in the text mining process from Figure 1. These steps include feature selection and pattern discovery. Topic modeling tries to find topics that are contained in the corpus.

Latent Dirichlet allocation (LDA) is a generative probabilistic model that assumes that each document in the corpus is a random mixture of different topics, and each topic is characterized by a distribution over words. In other words, the corpus contains unknown topics, that are spread out in multiple documents and each topic is characterized by a group of words. Words can also belong to multiple topics with varying probabilities. (Blei, 2012; Blei et al., 2003)

LDA improved upon earlier models by allowing each document in the corpus to contain multiple topics to varying degrees (Blei et al., 2003). Earlier models were limited by only allowing each document to be part of only one topic. LDA allows for example modeling of course evaluation surveys with open feedback, since it is likely answers to open questions will contain multiple topics in a single document.

LDA does not know how many topics there are in the corpus. Instead, the topic count defined by the user beforehand, meaning LDA always generates as many topics as is specified. There have been solutions for finding the best amount of topics like running the LDA multiple times with different topic counts and optimizing the perplexity of the model, although the best measurement for a topic modeling algorithm is interpretability by humans, which cannot be calculated (Blei, 2012; Wang and Goh, 2020).

LDA returns the words and the probabilities that they belong to a specific topic, but it does not return the labels of the topics. Instead, understanding what the topics are about is a human task of interpretation. There have been a few attempts in automatically naming the topics generated by topic modeling.

Phan et al. used a trained classifier to classify the topics generated by LDA into multiple different categories. They used two corpora as the input for LDA, one from Wikipedia and one from MEDLINE. The classifier was trained with separate data. With MEDLINE, for example, the goal was to categorize abstracts into certain diseases, and the classifier managed to do that with 66% accuracy. With Wikipedia, they used predefined categories

such as “business” and “computers” to categorize Wikipedia articles. This accuracy was much higher at 84%. (Phan et al., 2008)

Hindle et al. used LDA to categorize commit messages from three large relational database management systems and they trained a classifier to name these topics as different non-functional requirements. They found out that the topics can be given labels using semi-unsupervised methods, but supervised methods perform better. Both methods yield results which are much better than randomly assigning labels to topics, although they aren’t exactly accurate either. (Hindle et al., 2013)

Using machine learning methods to name the topics generated by topic modeling requires that the topics are specified beforehand with examples to train the algorithm. Since both LDA and STM require the topic count beforehand, the topic labels can be created when the optimal topic count is tested and selected. This can be done for example with a subset of the corpus, or training the model with the current dataset, so that new datasets of similar type can be categorized and labeled using the same topic count and trained classifier. This makes the topic modeling and classifier specific to the selected type of documents, and not easily generalizable. There is also the issue that after the topics are selected and labeled, new topics cannot be identified and labeled correctly, since they have not been taught to the model.

Structural topic model (STM) improves upon LDA by including document-level metadata in the analysis. In addition of taking in the bag-of-words representation of the corpus, STM can also take in document-level covariates. This means that for example in surveys, quantitative data like gender of the respondent can be included as a covariate in the model. Roberts et al. demonstrated that including covariate information does account for better results as the variance in topic prevalence is reduced. (Lucas et al., 2015; Roberts et al., 2019, 2016)

Another improvement of STM over LDA is the explicit estimation of correlation between topics. In other words, STM estimates how different topics relate to each other. This allows

for visualization of the topic correlations, which can be useful for getting a deeper understanding of the corpus-level structure of the topics. (Lucas et al., 2015)

While STM is an extension to LDA, it is not built directly on top of LDA. Instead STM combines and extends three models: correlated topic model (CTM), Dirichlet-multinomial regression (DMR) topic model and Sparse additive generative (SAGE) topic model (Roberts et al., 2013). CTM builds on top of LDA by allowing correlations between the topics. Correlations between topics are achieved using logistic normal distribution, instead of Dirichlet distribution (Blei and Lafferty, 2006). DMR topic model allows the inclusion of arbitrary meta-data in the model to improve the generation topics (Mimno and McCallum, 2008). SAGE is a multifaceted generative model, meaning SAGE can use multiple different probability distributions without having to switch between them to draw words into topics (Eisenstein et al., 2011). SAGE is used to include topic, covariate and topic-covariate interaction in the word's distribution in STM (Roberts et al., 2013).

2.2.4 Interpretation of topic models

The last step of the text mining process as depicted in Figure 1 is interpretation. This human task means understanding the generated topics and what can be interpreted from them. Commonly in literature, LDA and other topic models have been visualized by listing the topics and the most relevant words for that topic. This visualization can be seen for example in (Hu et al., 2019; Sliusarenko et al., 2013; Wang and Goh, 2020). Since topic models do not know the names of the topics, naming the topics is the main interpretation activity of understanding the results. In this sense, topic models create summaries for the topics found in the text, even though the summaries are just individual words and documents relating to that topic.

Since STM allows for correlations between topics, this can also be visualized by creating a map of topics with correlations between them indicated by the width of the line. This is demonstrated for example in (Hu et al., 2019; Lucas et al., 2015). Visualizing the topics and their correlations allows for deeper understanding of the corpus, especially as these relations might be very hard to pick up from the text by manual coding.

STM can include outside variables in the model and visualizing these variables and their relations to the topics might yield interesting results. For example, Hu et al. visualized topics from hotel reviews and how they relate to the overall rating of the review. This showed for example that topics “dirtiness” and “severe service failure” were found more often in negative reviews, while topics “decent location” and “staff attitude” were found more in positive reviews (Hu et al., 2019). Similar visualizations can be done, for example, with political analysis by visualizing topics by conservative-liberal axis (Roberts et al., 2019).

2.2.5 Sentiment analysis

Sentiment analysis is a text mining method used to understand the feelings or thoughts of the writer from the text (Tedmori and Awajan, 2019). Earlier methods categorized documents or individual sentences into either positive, negative or neutral, but current methods categorize sentiments based on the aspect they are expressed towards (Tao and Fang, 2020). This is called aspect-based sentiment analysis (ABSA).

Sentiment analysis can be done on three levels: document, sentence and entity or aspect (Hu and Liu, 2004). Documents can contain multiple different sentiments. For example, in a course evaluation survey answer, student might complain about group work being difficult while also praising the lecturer for explaining the subject well. In this case, it is hard to assign a positive or negative sentiment to the document. This problem continues in the sentence level as multiple differing sentiments can be also expressed in a single sentence, for example “The lectures were great but too long”. In this case “lectures were great” is a positive sentiment, but “lectures were too long” is a negative sentiment, and both sentiments focus on the same target “lectures”. Therefore, it makes sense to analyze sentiments on the entity or aspect level; otherwise all the expressed sentiments cannot be accurately identified. ABSA is especially useful in understanding product reviews, since reviewers usually focus around specific aspects of the products (Hu and Liu, 2004).

Sentiment analysis follows mostly the same steps as text mining in general. Text mining steps are shown in Figure 1. After preprocessing of the text, the next step is feature

extraction. Feature extractions can be done either using lexicon-based approaches or statistical approaches. Lexicon-based methods use a lexicon of words that are used to identify the relevant words from the text. Statistical methods, on the other hand, do not use lexica and are instead based on algorithms that discriminate between important and unimportant words for the semantic analysis. (Tedmori and Awajan, 2019)

Sentiment classification is the next step after feature extraction in sentiment analysis process. In sentiment classification words or pieces of text are categorized into classes like “positive”, “negative” and “neutral”. There is a division of three major ways of doing sentiment classification: lexicon-based, machine learning and hybrid approach. Lexicon-based approaches use lexicons to categorize the pieces of text into the selected classes, whereas machine learning approaches use trained models to categorize the sentiment behind the pieces of text. Hybrid approaches combine both of the methods and these approaches have been the most popular in literature. (Tedmori and Awajan, 2019)

The final step is visualizing or summarizing the results to the user, Tedmori & Awajan call this step sentiment summarization. Summarization is dependent on the topic as, for example, timelines can be used to show changes in overall sentiment over time, while product reviews can be summarized by listing ratings of the different aspects of the product. (Tedmori and Awajan, 2019)

Khoo & Johnkhan compared six sentiment lexicons: General Inquirer, Multi-perspective question answering (MPQA) subjectivity lexicon, HU & Liu opinion lexicon, National research council Canada (NRC) word-sentiment association lexicon, Semantic orientation calculator (SO-CAL) lexicon and WKWSCI sentiment lexicon, which they developed themselves. All these lexicons were coded by hand. Lexicons were tested with a dataset of product reviews and a dataset of news headlines. Overall Hu & Liu, WKWSCI, MPQA and SO-CAL did the best on product reviews, with accuracies around 75% in predicting the sentiment of the review. In news headlines WKWSCI, NRC and General Inquirier did the best, with accuracies around 65%. (Khoo and Johnkhan, 2018)

2.2.6 Emotion analysis

Sentiment analysis is done to categorize sentiments into two categories “negative” and positive”, or three categories “negative”, “positive” and “neutral”. In addition to sentiments, emotions can be also identified from text, like sadness, anger and joy. Emotion analysis follows the same procedures as sentiment analysis, but emotion analysis has a different classification goal. Identifying sentiments and emotions from text are treated as separate problems, although sentiments can be identified from the emotions (Kumar et al., 2019).

Detecting emotions is done using lexicons. These lexicons consist of words and their labeled emotions, and they can be used as input for machine learning classification algorithms. Wang et al. created a large dataset for 7 basic emotions (joy, sadness, anger, love, fear, thankfulness, surprise) by collecting and analyzing tweets using the hashtags to identify the emotion that is expressed in the actual tweet. The example they give is a tweet “I hate when my mom compares me to my friends. #annoying”, where the tweet is labeled under “anger”, since the hashtag “annoying” is interpreted as being sub-category for “anger” (Wang et al., 2012). Koto & Adriani used similar methods of coding tweet sentiments with the hashtags to create four emotion lexicons from Twitter each with eight emotions (joy, trust, sadness, anger, surprise, fear, anticipation, disgust) commonly called Plutchik’s wheel (Koto and Adriani, 2015).

Distributional thesaurus is a system for finding synonyms for words where the related words are ranked by their similarity (Biemann and Riedl, 2013). Kumar et al. used a distributional thesaurus to expand the lexicon of their model by allowing it to recognize words similar to the base emotion words, overall improving the emotion analysis performance, thus highlighting the importance of the lexicon in emotion analysis. The overall goal was to do sentiment analysis through emotion analysis and it worked well, therefore empirically validating the connection between emotion and sentiment (Kumar et al., 2019).

Tabak & Evrim compared emotion lexicons and their effects on emotion analysis. These lexicons included National research council Canada (NRC) word-sentiment association

lexicon, EmoSenticNet (ESN), DepecheMood (DPM) and Topic based DepecheMood (TDPM). The lexicons contain different emotions and words based on those emotions, for example NRC contains the eight emotions from Plutchik's wheel and two sentiments (positive, negative), while ESN contains six emotions (joys, sadness, disgust, anger, surprise, fear), and DPM and TDPM are built with eight emotions (happy, sad, angry, afraid, annoyed, inspired, amused, don't care). For comparison, matching emotions were selected from NRC and ESN, while DPM and TDPM were mapped to match the emotions of NRC and ESN. Overall NRC and DPM performed the best in classifying emotions form news headlines. (Tabak and Evrim, 2016)

2.3 Visualization

Visualization is communicating information in an efficient way to human observers. There are guidelines about how to do visualization correctly, but they are specific to a certain context, and no universal correct solution exists. Engelke et al. proposed a process model for creating a database for visualization guidelines, although it has not been taken further than that. (Engelke et al., 2018)

A universal guideline for creating visualization was proposed by Shneiderman. He summarized his guideline in what he calls information seeking mantra (ISM): "Overview first, zoom and filter, then details-on-demand" (Shneiderman, 1996). ISM has been called influential by, for example, (Craft and Cairns, 2005; Engelke et al., 2018; Kandogan and Lee, 2016). The first step "Overview first" means showing the data in its whole to the user (Shneiderman, 1996). The overview allows the user to get an overall feeling for the data and notice relationships between the components of the data and patterns that might exist (Craft and Cairns, 2005). Zooming allows the user to look at points of interest at a more fine-grained level and filter out unnecessary information by navigation (Craft and Cairns, 2005; Shneiderman, 1996). Filtering accomplishes similar results as zooming, but the reduction in complexity happens by removing unnecessary data points, so that the user can select points of interest (Craft and Cairns, 2005). Details-on-demand allows viewing detailed information about individual data points, which in practice usually means showing additional

information by hovering or selecting a data point or a group of data points (Shneiderman, 1996). Since details-on-demand does not change the current view of the data, it makes it possible to solve specific tasks quickly (Craft and Cairns, 2005).

Additional steps in the ISM are “relate”, “history” and “extract” (Shneiderman, 1996). While they are not part of the “Overview first, zoom and filter, then details-on-demand”, they are still relevant to the ISM. Relate refers to allowing users to find relationships between data points by highlighting or filtering to show the related data points (Shneiderman, 1996). History means allowing the user to undo their actions to go back to a previous state (Shneiderman, 1996). Allowing users to return to previous states easily makes data exploration much easier and faster (Craft and Cairns, 2005). Finally, extract means allowing the user to save their work and extract it from the software as a file, since it is likely needed again later or in a different context, and the file can be shared with others (Craft and Cairns, 2005; Shneiderman, 1996).

Even though ISM is widely used, the original paper does not provide great explanations about the steps and the reasons behind them. Therefore Craft & Cairns conducted a literature review to see how ISM has been used. Multiple papers used ISM as a guide in their own visualization implementation, even though usually there was no rationale behind why ISM was selected, or it was not specifically mentioned how the ISM was used. Overall the ISM does not provide step by step answers, instead ISM only offers practical advice. While this advice has been deemed useful, it would make sense to build more detailed guides on top of the ISM, and verify the scientific validity of ISM. (Craft and Cairns, 2005)

Kelleher & Wagener listed their own ten guidelines for creating visualizations based on literature. These guidelines are meant for scientific plots unlike Shneiderman’s guidelines which are more geared towards interactive visualization programs. Each guideline is based on a scientific study, and the guidelines are meant as general principles, but there might be exceptions to every guideline. The guidelines are listed below.

1. Create the simplest graph that conveys the information you want to convey.
2. Consider the type of encoding object and attribute to create a plot.

3. Focus on visualizing patterns or on visualizing details, depending on the purpose of the plot.
4. Select meaningful axis ranges.
5. Data transformations and carefully chosen graph aspect rations can be used to emphasize rates of change for time-series data.
6. Plot overlapping points in a way that density differences become apparent in scatter plots.
7. Use lines when connecting sequential data in time-series plots.
8. Aggregate larger datasets in meaningful ways.
9. Keep axis ranges as similar as possible to compare variables.
10. Select appropriate color scheme based on the type of data.

While meant for scientific plots, these guidelines work well for creating plots for more regular data visualization, as these guidelines tend to focus around making the visualization as clear and easy-to-read as possible. (Kelleher and Wagener, 2011)

Visualization evaluation is a separate task from visualization. Even when guidelines are being followed, the results should be evaluated with the actual users. Since visualization can only be tested with users or experts, Sousa Santos & Dias list multiple best practices for the evaluation tasks. These best practices include, for example, using several evaluation methods whenever possible and doing heuristic evaluations before moving to testing with actual users. (Sousa Santos and Dias, 2013)

Corell et al. brought up the point that visualization is dependent on the variables selected for the graphs, and in case of density plots, histograms and dot plots it is possible to make errors (spikes, outliers, gaps) in the data disappear from the visualization. Using more bins in histograms, less smoothing in density plots and more transparency in dot plots alleviate this issue by making the errors in the data more noticeable. This is especially important in exploratory data analysis, where these kinds of plots are usually used as sanity checks. (Correll et al., 2019)

As mentioned in the section 2.2.4, topic models can be visualized by listing the words in order of importance for the topics. This can be enhanced by visualizing the word relevance to the topic by using bar graphs, which can be seen, for example, in (Roberts et al., 2014) or in the example Figure 2 from (Robinson, n.d.). An R package for STM also allows for creating word clouds for each topic (Roberts et al., 2019). To get the details-on-demand as suggested by (Shneiderman, 1996), the R package also allows to retrieve documents with high association to a specific topic as to give more context to what the topic might be about (Roberts et al., 2019). Following ISM, relations can be visualized by plotting the topics as a graph of connected nodes, where each topic is a node and the connection is based around the strength of the correlation (Hu et al., 2019; Roberts et al., 2019). Figure 3 contains topic correlation map of the topics identified from hotel reviews by (Hu et al., 2019) as an example visualization. The relations between topics and document covariates can be visualized as a scatterplot where topics are placed on the plot based on how much they correlate to a specific polarity of the outside covariates (Roberts et al., 2019). Figure 4 by (Roberts et al., 2019) shows an example of visualizing covariate topic relations in political analysis.

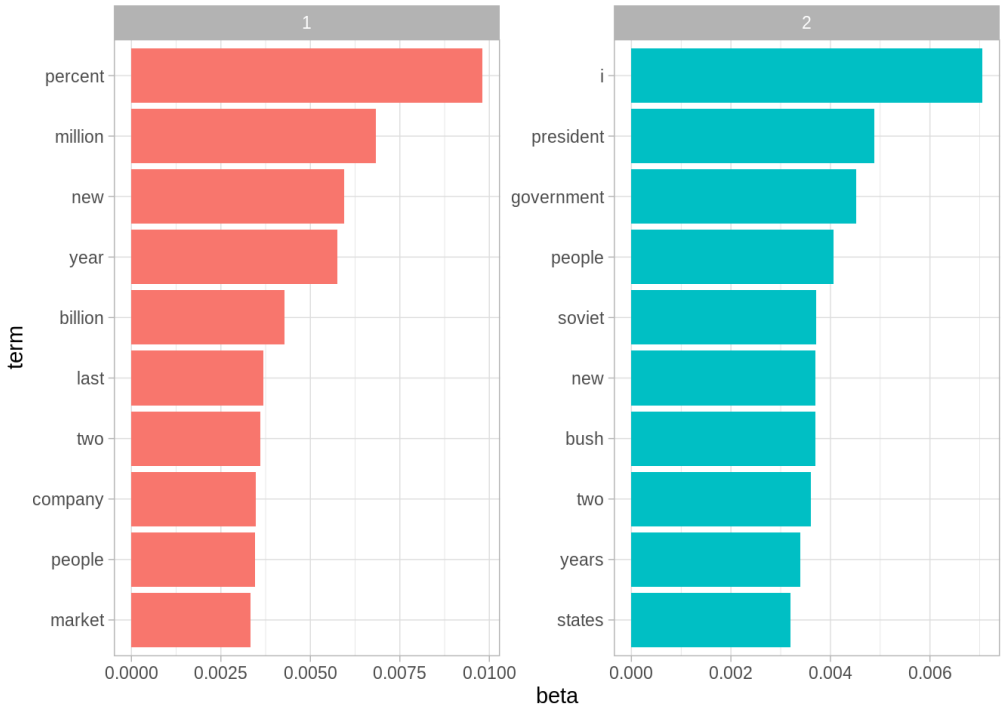

Figure 2. Bar graph visualizing word relevance for two topics

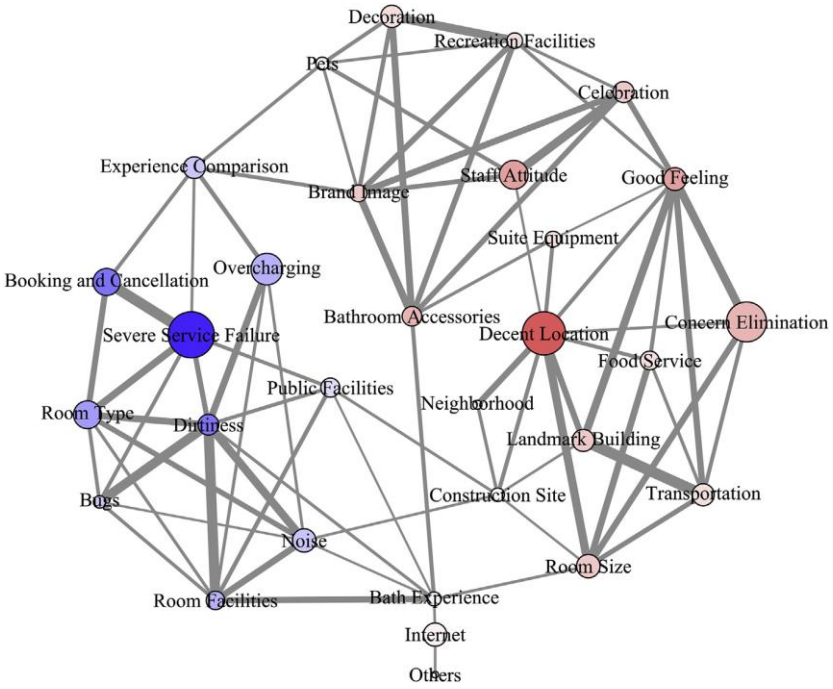

Figure 3. Topic correlation node map

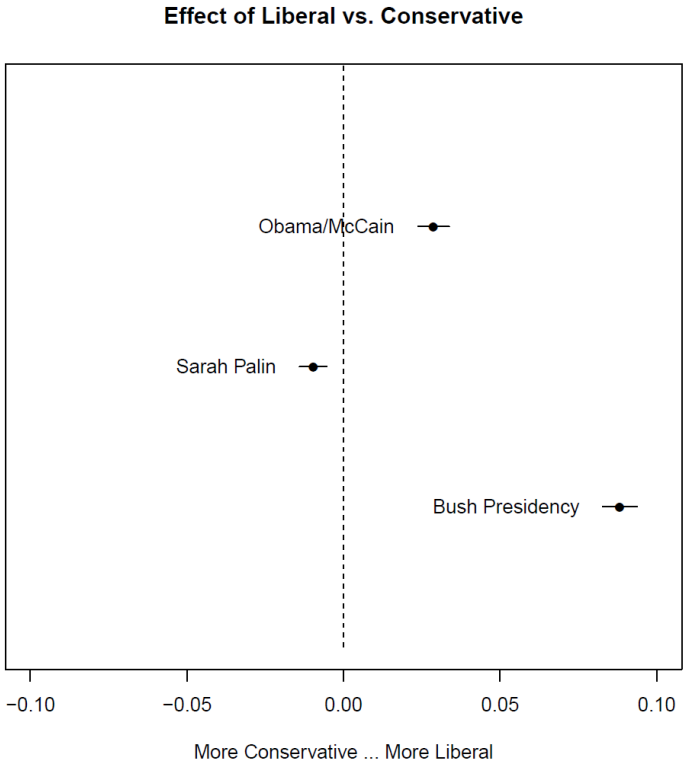

Figure 4. Topic covariate relation plot

Sentiment analysis is usually visualized with word clouds and line charts, while other less common methods are parallel coordinate plots, maps, pie charts, bar graphs and histograms (Almjawel et al., 2019). Word clouds are used to show the most relevant words, their sentiment and the count of the words in the data, as seen, for example, in (Almjawel et al., 2019; Healey and Ramaswany, 2019). Line graphs are usually used to show changes in the sentiment over time, as seen in (Almjawel et al., 2019; Da Silva Franco et al., 2019; Healey and Ramaswany, 2019).

Healey & Ramaswany have created an online tool for visualizing emotions in tweets called Sentiment Viz. The tool allows user to specify keywords to fetch recent tweets. The tweets are then analyzed and the results are visualized (Healey and Ramaswany, 2019). Emotion in the tweets is visualized using Russell model of affect (Russell, 1980) as shown in Figure 5. Russell model of affect is a two-dimensional wheel of emotions where the axes are from unpleasant to pleasant and from subdued to active. Other emotions are a varying combination of emotion in the axes and are thus placed on the outer ring of the wheel. For example, excited is at a 45% angle between pleasant and active. Other emotion visualization methods included in the Sentiment Viz site are a heatmap showing the count of different emotions on the Russell model of affect and a graph showing four word clouds with words that are tagged to the four quadrants of Russell model of affect (upset in upper-left, happy in upper-right, relaxed in lower-right, unhappy in lower-left) (Healey and Ramaswany, 2019). Sentiment Viz also includes a timeline which shows the change in the four basic emotions in Russell model of affect over time as a bar graph where the emotion is visualized using color (Healey and Ramaswany, 2019). Sentiment Viz tool was used by (Caballero et al., 2018) to study tweets relating to a university.

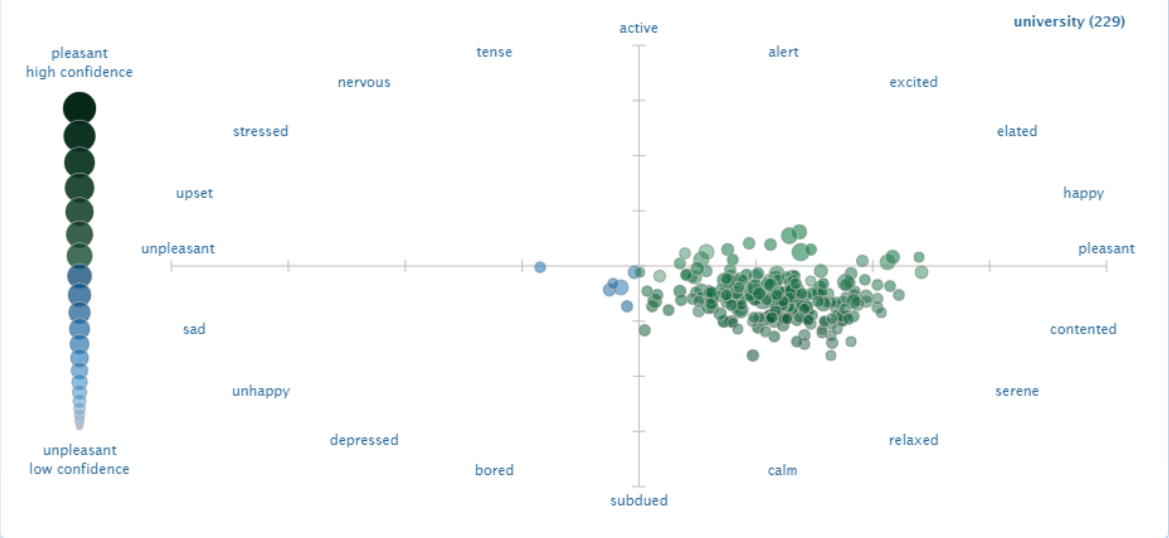

Figure 5. Sentiment Viz Russell model of affect for the keyword "University"

Da Silva Franco et al. created a tool called UXmood to visualize user emotions from a video to aid in the user experience development and testing. They used a timeline to show the emotions during a specific time, and a word cloud with words categorized with colors based on the emotion they were most used with to summarize the whole video. More specific to the video context was a chronological animation scatterplot that showed where the user was looking at on the screen and what kind of emotion their face was expressing at that time. (Da Silva Franco et al., 2019)

2.4 Summary of related work

Text mining has been and continues to be important field of research, with multiple different techniques and goals. Topic modeling and sentiment analysis are used successfully in multiple different domains where the goal is to generate information for humans. Visualization has a lot of guidelines but no systematic solution for designing visualization exists yet. The overall literature review is summarized in Table 1 below.

Table 1. Summary of literature

Authors	Findings
Text mining	
Ahonen et al., 1997	Episode and episode rule techniques have potential in text mining
Chowdhury, 2003	Current NLP methods show promise, while still not being good enough for wide implementation in the industry
Feldman et al., 2003	Using LitMiner for finding and visualizing biomedical data
Korenius et al., 2004	Lemmatization yields better results for Finnish language than stemming in clustering
Romero and Ventura, 2007	Educational data mining is a promising upcoming field, especially with the rise of e-learning systems
Sanchez et al., 2008	A proposal of dividing text mining into text data mining and text knowledge mining
Jordan, 2011	Answers to quantitative and qualitative parts of student course evaluations correlate weakly, but information gained from text mining student course evaluations can be used on institutional level
Biemann and Riedl, 2013	An implementation of a distributional thesaurus
Fokkens et al., 2013	Highlights the impact of different text preprocessing steps by trying to reproduce earlier studies
Sliusarenko et al., 2013	Answers to quantitative and qualitative parts of student course evaluations correlate only partly
Chen et al., 2014	A methodology for using social media data to gain insights about students' experiences
Tucker, 2014	Student course evaluations provide useful information and students do not abuse their anonymity to harass the course teachers with course evaluations
Hashimi et al., 2015	A set of criteria for the selection of appropriate text mining method
Agaoglu, 2016	Data mining techniques on student course evaluations can be used to evaluate the course teacher effectively
Stupans et al., 2016	Demonstrates using text mining tool Leximancer on student course evaluations
Zhang et al., 2017	Empirical evidence of the usefulness of clustering methods in student course evaluations
Eler et al., 2018	Visualizes the effects of text preprocessing in text mining
Gottipati et al., 2018	Decision trees work best for extracting suggestions from answers to open questions in student course evaluations from the tested methods
Topic Modeling - LDA	
Blei et al., 2003	LDA algorithm for topic modeling
Phan et al., 2008	Automatically naming the topics generated by LDA
Biggers, 2012	LDA performance is mostly unaffected by text preprocessing steps in the domain of software source code
Blei, 2012	A general explanation of LDA
Hindle et al., 2013	Automatically naming the topics generated by LDA
Curiskis et al., 2020	A comparison of document clustering and topic modeling methods on social media data
Wang and Goh, 2020	Dimensions of gameplay experience and their importance to the players mined from online game reviews
Topic Modeling - STM	
Blei and Lafferty, 2006	CTM algorithm for topic modeling
Mimno and McCallum, 2008	DMR algorithm for topic modeling
Eisenstein et al., 2011	SAGE algorithm for topic modeling
Roberts et al., 2013	STM algorithm for topic modeling
Roberts et al., 2014	Demonstrates how STM can be used with open ended responses

Lucas et al., 2015	Examples of how to use STM to compare political texts
Roberts et al., 2016	Demonstrates STM
Hu et al., 2019	10 topics that manifest in negative hotel reviews and how the topics differ in high-end hotels compared to low-end hotels
Roberts et al., 2019	Demonstrates how to use STM R package
Sentiment analysis	
Hu and Liu, 2004	A set of techniques for feature-based summaries of product customer reviews
Mohammad and Yang, 2013	New word-emotion lexicon, information about how men and women use words with different emotions in emails
Koufakou et al., 2016	Demonstrates successfully using text mining methods on the open question responses of student course evaluations
de Paula Santos et al., 2016	A model of educational data mining for evaluating teaching practices
Pong-Inwong and Kaewmak, 2016	Voting ensemble method is efficient in sentiment analysis
Khoo and Johnkhan, 2018	Introduces new sentiment lexicon WKWSCI that outperforms existing sentiment lexicons in non-review texts
Stojanovski et al., 2018	A system that outperformed other state-of-the-art methods in analyzing Twitter messages
Tseng et al., 2018	Classifiers that consider time series factors (RNN, LSTM, attention RNN) perform better in sentiment analysis than those that do not consider time series factors.
Ahmad et al., 2019	Demonstrates using sentiment analysis on student course evaluations for evaluating course teacher performance
Almjawel et al., 2019	Demonstrates the visualization of sentiment analysis on Amazon book reviews
Kumar et al., 2019	A neural network that performs sentiment analysis through emotion analysis and outperforms current state-of-the-art systems
Tedmori and Awajan, 2019	Different use cases and methods of sentiment analysis
Tao and Fang, 2020	Aspect enhanced sentiment analysis
Emotion analysis	
Wang et al., 2012	An emotion lexicon made from 2.5 million tweets
Koto and Adriani, 2015	Four emotion lexicons made from Twitter
Tabak and Evrim, 2016	A comparison of different emotion lexicons and their effects on emotion analysis
Caballero et al., 2018	Demonstrates using tweets to analyze the perception of an institutional organization
Da Silva Franco et al., 2019	A tool for performing emotion analysis on user testing
Healey and Ramaswany, 2019	SentimentViz, an online tool for performing emotion analysis on tweets
Visualization	
Russell, 1980	A model of affect (Not about visualization, but was used as a source by other visualization papers)
Shneiderman, 1996	ISM, a guideline for visualization, taxonomy of visualization by data type
Craft and Cairns, 2005	ISM is widely used, but it is usually not specified how it is used to guide visualization
Kelleher and Wagener, 2011	10 guidelines for creating scientific visualizations based on literature
Sousa Santos and Dias, 2013	Best practices for evaluating visualization methods
Kandogan and Lee, 2016	Suggests that a systemic approach to visualization design is required
Engelke et al., 2018	A conceptual model for supporting the definition, curation and communication of visualization guidelines
Correll et al., 2019	Demonstrates how common visualizations of distributions can hide errors in the data and recommends best practices for avoiding hiding flaws in the data

3 RESEARCH METHOD

Design has been defined as “The conception and planning of the artificial” (Buchanan, 1992). It is the planning required to create something new that did not exist in the world before. Design is inherently a wicked problem (Rittel and Webber, 1973).

Wicked problems are described as problems, where the problem cannot be clearly stated, there is no exhaustible set of potential solutions (as there are too many) and it is unclear when the solution is reached since it cannot be tested (Rittel and Webber, 1973). This definition of the wicked problem is a summary as the whole definition consists of ten qualities of wicked problems that were listed by (Rittel and Webber, 1973). Rittel & Webber originally tied the wicked problems to planning policies on a societal level, in other words to social sciences, but Buchanan argued that all but the most trivial design problems are inherently wicked problems (Buchanan, 1992). Designing requires the creation of something new and since it does not exist yet, the scope of the potential solutions is unlimited, and this limitlessness makes it impossible to know if the optimal solution was discovered (Buchanan, 1992).

Rittel & Webber separated solving wicked problems from the scientific method as the problem cannot be stated clearly and it cannot be known if the solution solved the problem, which is in contrast to the natural sciences where the problem can be unambiguously stated and its solution measured and confirmed (Rittel and Webber, 1973). Farrell & Hooker argue that all the ten characteristics of wicked problems can be reduced to three main sources: finitude, complexity and normativity, and that the natural sciences deal with the exact same issues (Farrell and Hooker, 2013). There is no clear division between wicked and tame (tame problem being the opposite of a wicked problem (Rittel and Webber, 1973)) problems, instead every problem is on a scale between the two extremes (Farrell and Hooker, 2013). Most importantly, science and design deal with the same issues of finitude, complexity and normativity with varying degrees, meaning the separation of design from science is invalid (Farrell and Hooker, 2013). Even more so, design and science share a core cognitive process and they cannot be separated based on that either (Farrell and Hooker, 2014).

Design science research (DSR) is based around solving wicked problems by creating artefacts, and the knowledge about the problem and its solution is acquired by the design and use of the artefact (Hevner and Chatterjee, 2010). Similarly, DSR in information systems (IS) also deals with wicked problems. The problems faced in IS have unstable requirements, complex interactions within the system and outside the system, there are no static processes for the creation of the artefact and the creation of the effective solutions is dependent on human cognitive and social abilities (Hevner et al., 2004). So, DSR in IS deals with problems by designing new artefacts that are specific to a problem domain and generates new knowledge based on the artefacts (Hevner and Chatterjee, 2010). The artefacts in IS can include, for example, software tools, frameworks, design patterns and protocols.

Hevner lists seven guidelines for conducting design science research:

- DSR requires the creation of a purposeful artifact
- The artefact is created for a specified problem domain
- The utility of the artefact must be evaluated
- Design and creation of the artefact must provide contributions to the areas of DSR
- The artefact must be constructed and evaluated with scientific rigor
- Designing the artefact is a search problem of finding an effective solution from the problem space
- The results must be communicated effectively

These guidelines give direction to conducting DSR, but it is up to the researcher to figure out how to apply them correctly and how thoroughly each guideline should be followed (Hevner et al., 2004).

This thesis uses the DSR as its primary research method. The artefact is created to gain useful insights from student evaluation of teaching data. The utility of the artefact will be evaluated based on the research questions laid out in section 1.1. This thesis contributes to the design science research by providing a novel artefact while also demonstrating the usability of text mining techniques on student course evaluations. The artefact combines already existing

scientifically evaluated text mining methods, while also using evaluation methods based on literature, thus demonstrating the use of rigorous research methods. The artefact is searched by iteratively designing it, while comparing it to the set goals, while also using existing literature as a starting point for development. Finally, this thesis communicates the results of this research, thus fulfilling the guidelines of DSR.

In practice, the design science research method process by (Peffers et al., 2007) is used. The process model can be seen in Figure 6 by (Peffers et al., 2007). As the problem was already identified as the topic for this thesis, the process is started by defining the objectives of a solution. This is done in section 1.1 as was mentioned above. The artefact design is presented in chapter 4 followed by the details of implementation in 5. Demonstration is done in the same chapter as evaluation, which is the chapter 6.

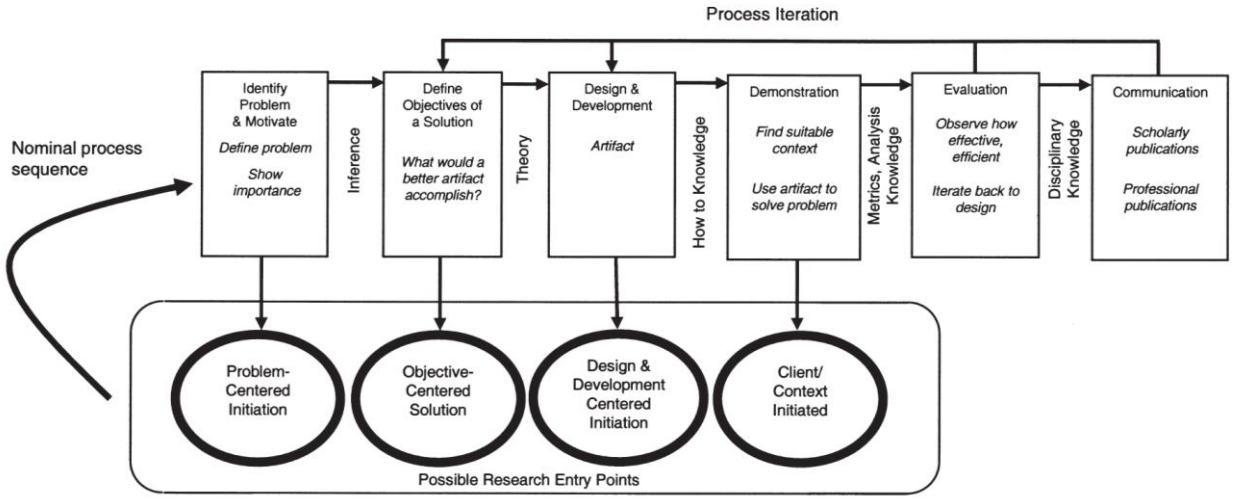

Figure 6. Design science research methodology process model

4 ARTEFACT DESIGN

When it comes to evaluation surveys, qualitative open-ended questions have the advantage over Likert-type questions by allowing the respondent more freedom in their answers, in addition of allowing answers that were not expected in the survey design (Vinten, 1995). This is especially useful in course evaluation surveys, where the open questions allow the respondent to point out individual pain points or positive aspects of the course. This information is more useful in improving the course than just receiving lower ratings on a Likert-scale. That is not to say that closed questions are useless, since they are helpful in getting the overall understanding about whether the feedback is positive or negative. Closed questions give a direction, but they only provide as detailed information as is specifically asked in the question. Asking about all the problems that might occur during a course would make the survey unreasonably long, and mostly likely the response rate would drop drastically. Thus, the best balance is achieved using a mixture of closed and open-ended questions in course evaluation surveys.

The issue with open-ended questions is that they require human interpretation, and especially coding of the answers is a laborious task (Vinten, 1995). This is not an issue with courses with about 30 students, but interpreting the feedback becomes very costly and unreasonable as the course participant count rises to hundreds, as is the case in some mandatory freshmen courses in LUT University. The other issue with open-ended questions is that they require more effort from the respondents and thus the answer rates are usually between 10% and 60% in course evaluation surveys (Jordan, 2011). Effort required to answer open-ended questions is also visible from the broadness of the question. Broader the question, higher the effort required to answer it, as formulating the answer becomes more difficult. Therefore broader questions tend to receive longer answers, but less answers overall than more specific questions that receive more answers, but the answers are shorter in length (Jordan, 2011).

Jordan also raises the point that open-ended questions are still usually specific to a certain aspect of the course performance, and these kinds of questions receive the least answers overall. Fully open questions, for example “Additional comments”, tend to receive higher

amount of answers than more specific questions. This is likely due to students having thoughts and opinions that do not fit in any of the specific questions. (Jordan, 2011)

Open-ended questions tend to receive twice as many positive answers than negative answers. The positive answers also tend to be more general in nature, whereas negative answers tend to focus around more specific pain points. (Alhija and Fresko, 2009; Jordan, 2011)

There is also the disparity between closed Likert-type questions and open-ended questions, where closed questions might receive a good mean value, but the open evaluation is critical, or vice versa. The difference likely caused of Likert-type questions not addressing the issue that the student then addressed in more open questions. (Jordan, 2011)

4.1 LUT course evaluation survey structure

The student course evaluation surveys used in LUT university are constructed from questions that are scientifically validated in student evaluation of teaching literature. There have been multiple iterations of the structure of the survey and the questions used in it, especially as the practice of collecting student feedback started as early as 2004 in the LUT university. As such, surveys from different years have different questions. For example, surveys used in the years 2016-2017 have 17 questions that are included in all the surveys, while surveys used in the years 2017-2018 have only 5 mandatory questions. Teachers are also allowed and encouraged to add additional questions to the survey used in their course, so the surveys might not match one to one between the courses of the same year.

After 2017, the questions used in the survey have been about students evaluating their effort in the course, what factors affected their performance and how the teaching methods affected their learning. The goal is to make the students think about their part of the teaching as active learners, instead of just criticizing the used teaching methods. The focus is on open questions as they tend to be most informative for improving the courses, although both qualitative and quantitative questions are included in the survey.

4.2 Functional requirements

The goal of the artefact, as mentioned in section 1.1, is to surface meaningful information from a corpus to the user. To accomplish this, the tool must first allow the user to input the data. Then, the data must be preprocessed, analyzed and visualized to the user, so that the insights can be highlighted from the data. The functional requirements for the artefact are listed in the Table 2 below.

Table 2. Functional requirements

ID	Description	Reasoning
FR1	User can input data to the tool	Users should be able to insert the data they want to analyze
FR2	The tool works with Finnish and English	LUT offers courses in both Finnish and English, therefore the surveys and their results are also in Finnish or English
FR3	User can select the columns from the data for the analysis	LUT course survey is not static, so the tool should work with multiple different structures of data
FR4	User is presented with default options for using the tool	The tool should be easy and fast to use, so the user should be able to run the analysis immediately after uploading the data
FR5	User can modify the options used in the analysis	Topic modeling is highly dependent on the options, so the user should be allowed to tweak them for more accurate analysis
FR6	The tool summarizes text data	The goal of the tool is to make understanding the answers to open questions easier. This can be accomplished by creating summaries of the text data

FR7	The tool creates insights from data	The tool should create insights from the data that are hard to find by reading the texts
FR8	The tool allows for data exploration	Users should be able to explore the data in multiple ways
FR9	The analysis results are visualized to the user	Results should be visualized in multiple ways as a way to communicate them to the user
FR10	The user can filter the results	Giving more tools to users allows them to perform the actions they want. In other words, the tool is flexible

5 IMPLEMENTATION OF THE ARTEFACT

The artefact was named Palaute which is an acronym of the words plot, analyze, learn and understand topic emotions. Palaute is also Finnish for feedback. The source code is licensed as GNU general public license v3.0 (GPLv3) and can be found from (Grönberg, 2020).

The artefact is implemented using R programming language and R Shiny which is an R package for building web applications. Building the artefact as a web application makes it easy to use and accessible, since the users don't need to setup any environments to run the artefact nor do they require any skills in programming. This of course limits the tool to the programmed features which cannot be changed easily, although experienced users can download the source code and change it to match their needs. R Shiny allows the user to upload the data to the server for processing as is required in the functional requirement FR1 that is presented in the Table 2.

Functional requirement FR3 assumes that the data is in matrix form with rows and columns. The artefact thus works with comma-separated values (CSV) and allows the user to select what columns are used in the analysis.

To fulfill FR4 and FR5, the artefact has default options that allows the user to input the data and run the analysis without any additional tweaking. There are also options for manually setting the most relevant options for further analysis and data exploration.

FR6 and FR7 are accomplished by using topic modeling with sentiment and emotion analysis. Topic modeling finds topics from the text, thus creating a summary of the main themes. Sentiment and emotion analysis can be then applied to the documents in the individual topics generated by the topic modeling algorithm by finding all the documents that are related to that specific topic. This gives additional information about the topics. Sentiment and emotion analysis can also be applied to the whole data set to summarize the sentiment and emotions in the data. Sentiment and emotion analysis as well as text

preprocessing depend on the language of the text data. This makes it important that the user can select the language of the data to fulfill the requirement FR2.

FR8 is a combination of the requirements FR1, FR3, FR5, FR9 and FR10. Palaute allows data exploration by being flexible with the structure of the input file, allowing the user to customize the analysis options and giving the user tools for filtering the results. Visualization is essential to data exploration as communicating the results to the user in an effective way makes conducting exploratory data analysis faster and easier.

Palaute is a prototype that was created in two months by a single developer. There are likely some bugs, poor user interface choices and unclear behavior in the artefact. Due to the prototype nature of the artefact and time pressure, there is no formal software testing of any kind, meaning there are no unit tests for individual functions nor systems tests for the whole system.

5.1 Main functionality

The artefact performs topic modeling, sentiment analysis and emotion analysis on data sets of varying kinds. This core functionality of the artefact is built on two R packages STM (structural topic model) and Syuzhet. Topic modeling is done using the STM package by (Roberts et al., 2019). Sentiment and emotion analysis are done using the Syuzhet package by (Jockers, 2015). The Syuzhet package contains multiple different lexicons for performing sentiment analysis and NRC lexicon for emotion analysis. Syuzhet also allows using custom lexicons.

STM package contains a function for calculating the topic model. The topic model can be calculated using only the documents, but there can also be metadata in the form of covariates. Contribution of each topic to a document is called topic prevalence. The first type of covariates are prevalence covariates and they are used in the calculation of the topic prevalence. The second type of covariates are content covariates. Content covariates affect the words used in a topic and in the current implementation of STM content covariates create

strict groups of documents so that each document can only belong to a single group. (Roberts et al., 2019)

Prevalence covariates are external data to the documents, but they can be used in the calculation of topic prevalence. For example, in the context of course evaluation surveys a Likert-type question about the workload of the course can be used as a prevalence covariate. It is likely that surveys that rate the workload higher than average would also be more negative and thus the documents would covary with this data. A simpler example would be to use product reviews as the documents and a numeric rating of the product as the prevalence covariate.

Topical content covariates change the STM model a lot since the documents are forced into groups. It can be used to for example separate negative product reviews from positive reviews. In the context of course evaluation surveys it could be used with some Likert-type questions that would significantly affect the vocabulary used in the topics. The survey questions could also be included as content covariates as it would make sense that different questions are answered differently.

The artefact has support for using both covariate types, although the usefulness of content covariates is not fully realized since there are no visualizations or keywords for showing the differences between the document groups. As a limitation of the STM package, there can be only one content covariate, but multiple prevalence covariates are supported (Roberts et al., 2019). The artefact implements the support for multiple prevalence covariates and one content covariate.

STM package also contains tools for selecting the best model and the computationally best number of topics. A function which trains multiple models for each number of topics was implemented in the artefact. The user can specify the start and end values for the number of topics and multiple models are calculated for each value. It would be then up to the user to select the best model, but in the current version of the artefact this selection is automated. Best model is selected as a maximum mean of the normalized values of model’s semantic

coherence and exclusivity. While this might not yield the absolute best model, it should yield a good model (Roberts et al., 2014). The best model is the most interpretable by the user and there is no mathematical formula for this yet.

Sentiment analysis and emotion analysis are performed using NRC lexicon simply by matching the words in the data to the lexicon words and adding up the sentiment values for each matched word. This analysis does not consider the order of the words, the context of the words, negations nor emphasis, but it should still yield a general sense of the of the data.

The NRC lexicon used in the Syuzhet package was originally created in English, after which it was machine translated to multiple languages including Finnish. As machine translating individual words is not that difficult, the accuracy should be relatively high (Mohammad et al., 2016). The issue with using the NRC lexicon for Finnish analysis is that the words in the lexicon are mostly in their basic forms. Finnish is a highly inflected language and has thousands of inflections per word (Kettunen, 2005), meaning significant portion of the words tend not to be in their basic forms in Finnish texts. To improve the accuracy of the sentiment and emotion analysis, the Finnish NRC lexicon was stemmed using the same Snowball stemmer that is used by the artefact to do stemming in the text preprocessing step. English is still analyzed without stemming. The stemmed lexicon achieved about 6.5 times more total identified emotions and sentiments than the original Finnish NRC lexicon with the course evaluation survey data set specified in section 5.3. A more detailed evaluation of the effects of stemming the lexicon is shown in section 6.1.

The sentiment analysis and emotion analysis are performed on the whole data set as a summary of the corpus. For individual topics, representative documents are selected, and the sentiment and emotion analysis are run with only the selected documents. There are multiple ways of doing this selection of documents, but the current implementation is that the artefact selects the documents exclusively, meaning each document is added to the corpus of the topic that has the highest prevalence in that document. Dividing the documents exclusively among the topics makes sure that each document is used in the overall analysis only once, as multiple topics sharing the same documents would make the topics more similar with each

other. Another way of selecting the topic representative documents would be to select some arbitrary number of the documents that have the highest prevalence of a topic. This would mean that multiple topics can share same documents and it is no clear how many documents should be selected for each topic. Some of the documents that have an even distribution of topic prevalences would likely be left out of the analysis completely. Third way of building the topic corpora would be to select all the documents that have the topic prevalence over, for example, 50%. The issue of not using all documents is still present, but it is easier to argue for using documents with a topic prevalence of over 50% than using 50 documents with the highest prevalence for each topic.

The user has multiple options for performing the analysis. There is option for randomly sampling the data set for decreasing the analysis time, with a selection for the sample size. User can set the topic count and maximum iterations for the STM algorithm. There is also the option to set starting and ending values for the number of topics and let the algorithm decide the best model.

5.2 Preprocessing

The course evaluation survey data is in a format where there is a column for each question of the survey and each row is a respondent's answers to the survey. So, there are multiple columns that contain written answers to open questions. This means that the corpus of documents needs to be built from the data by combining the columns with the written answers to the open questions. To complicate this further, there are columns with data that is useless for the analysis like file name or language, so only some text columns should be used as documents. Finally, the user should be allowed to select what columns are used as covariates.

Building the corpus is done by creating a dropdown menu for each column of the original data set. From these dropdown menus, user can select the role for that column from four possible choices: document, prevalence covariate, content covariate and don't include. The artefact preselects these for the user by guessing the documents from columns containing text data, and by default selecting the first two numeric columns as prevalence covariates.

The corpus is then built by combining all the document columns into a singular column with selected covariates as separate columns.

By default, the STM package contains all the necessary text preprocessing required for the topic modeling task including lowercasing, removal of stopwords and special characters and stemming. The stemmer was underperforming in Finnish, so the stemmer was changed to Snowball stemmer. Snowball stemmer is purely algorithmic, but it tends to yield good results (Porter, 2001). While stemming is not as important in English, in highly inflected languages like Finnish Snowball stemmer reduces the word pool to about 20% of the original (Kettunen and Baskaya, 2011). Since it is common to remove rarely occurring words before topic modeling (Eler et al., 2018; Roberts et al., 2019), stemming helps in maintaining words that would be removed in their inflected form, as they would occur in that specific form too rarely in the corpus.

5.3 Visualization of the results

The results of the analysis are visualized in plethora of ways. The example plots in this section are generated with a data set of course evaluation survey answers from 2017 novice programming course in LUT university. The data set contains 104 answers to the survey with 10 questions, five of which are open questions and five which are Likert-type questions. The data set is in Finnish. Combining the open answers to a single column and removing all empty answers makes the data set 307 rows in size. STM model was trained with 11 topics, and a default value of 500 maximum iterations, during which the model converged under the default threshold of 1e-5 of relative word bound change in iteration 408. Two of the Likert-type questions were used as prevalence covariates. The sentiment and emotions were analyzed using the stemmed variant of the NRC lexicon.

Model convergence plot shows the model word bounds by iteration. This plot helps in the evaluation the model performance and helps in understanding how the model is fitted. Figure 7 shows the convergence plot for the data set. The model seems to find the basic structure of the data in about 10 iterations after which it slows down. An example in the STM vignette

shows much more logarithmic and smooth curve (Roberts et al., 2019). This can mean that the selected topic count of 11 does not represent the structure in the data well, or that the data is noisy and thus takes a long time to converge. This can be also seen in the ripples of the curve, as the model seems to converge but changes shortly after.

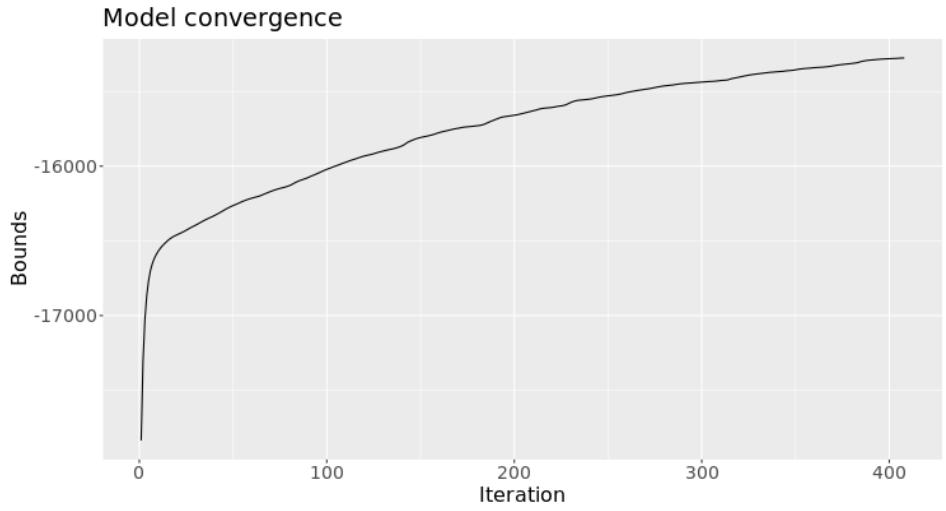

Figure 7. Model convergence

Topic visualization is done similarly to LDAvis by Sievert & Shirley. LDAvis uses Jensen-Shannon divergence to calculate the inter-topic distances, which are then reduced to two dimensions to be shown as a two-dimensional plot. Each topic is displayed as a circle, with the area of the circle being proportional to the topic proportion. (Sievert and Shirley, 2014)

Palaute adds to the LDAvis plot by expressing the sentiment of the topic as a color. This sentiment is calculated from the documents relevant to that topic. The inter-topic distances are calculated from the STM model's beta matrix, which contains the logarithms of the word probabilities by the topic. As STM uses logarithmic values of the word probabilities, opposed to just using word probabilities like in LDA, exponent function must be applied to the values in the beta matrix before the topic distances can be calculated using Jensen-Shannon divergence. The sizes of circles are proportional to the topic proportions, but this does not mean overlapping circles should be interpret as sharing similar words proportional to the overlap. Instead, distance between the topics is the measure of topic similarity,

meaning they use similar vocabulary. Another important note is that since the plot is a two-dimensional representation of a higher dimensional construct, information is lost as the distances are projected two-dimensionally. Dimensional scaling is done using classical multidimensional scaling. The dimension scaling algorithm tries to keep the inter-topic distance similar when reducing dimensions, but there is information that is lost. So, just because two topics are close to each other, it does not necessarily mean they should be merged as one, although this can also be the case.

Palaute allows for clicking on the topics in the topic distance plot to show additional information. This information includes the topic proportion as a percentage, the sentiment as a percentage and five keywords for each four methods of calculating keywords that are included in the STM package. Most probable keywords simply have the highest probability to belong to that topic, FREX keywords are calculated using frequency and exclusivity, lift keywords assign more weight to exclusive words by dividing the word frequency by their frequency in other topics, and score keywords are similar to lift but use the logarithmic frequency of the words in the calculation (Roberts et al., 2019).

Figure 8 shows the result of the model with 11 topics. There seems to be a cluster of topics 2, 7, 1 and 4, although topic 1 uses more negative language. Looking more closely at topic 2 shows that the keywords are about topics, interest and programming, while the keywords in topic 7 are about exercise classes. Figure 9 shows the additional topic information from topics 2 and 7. The keywords, like the data, are in Finnish and stemmed.

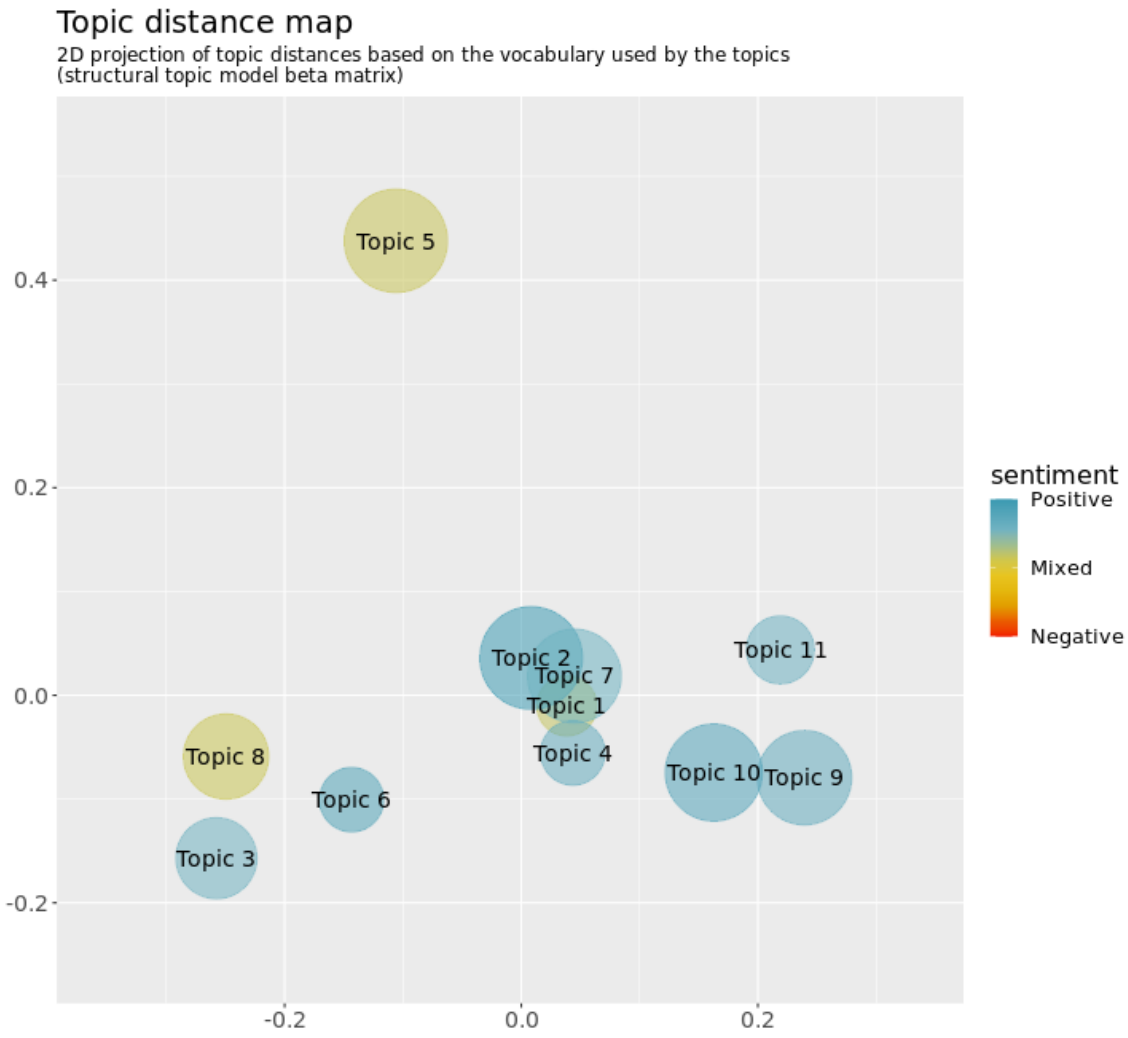

Figure 8. Inter-topic distances

Topic 2 Proportion: 12%

Sentiment: 98% positive

Keywords:
Most probable: palj aihe halus ohjelmoint kiinnostus

Frex: aihe kiinnostus suht halus kiinnostunu

Lift: aihe haast hyödyllis hyödyllisyys jakso

Score: aihe kiinnostus mahdollis kiinnostunu tulevaisuud

Topic 7 Proportion: 11%

Sentiment: 79% positive

Keywords:
Most probable: enem hyv sai harjoituks kuite

Frex: sai kuite pakollis enem harjoituks

Lift: edellis enit erilain erityis harkkatehtäv

Score: sai jois kuite kohd harjoituks

Figure 9. Comparison of additional information from topics 2 and 7

Theta matrix of the STM model contains the document topic proportions by topics. This matrix can be visualized to show what documents belong to which topics and how much of that document belongs to the other topics. In the artefact this is done by creating a scatter plot of the documents, where the color of the document is based on the highest topic prevalence, as is the size of the circle. So, larger circles have a larger portion of them dedicated to a single topic. Barnes-Hut variant of t-Distributed stochastic neighbor embedding (t-SNE) was used to dimensionally scale the data down to two dimensions (Maaten, 2014).

Documents that have similar topic proportions cluster together in this plot. When documents are highly cohesive in the sense that they belong mainly to one topic, it causes clear clusters of documents emerge in the plot to represent the topics. When the documents contain multiple topics more evenly, then the topics are not represented as single clusters. When the documents share similar topic proportions, they tend to share similar vocabulary, meaning semantically similar documents also cluster together. Topic labels are placed on the mathematical means of the document coordinates. The circles can be clicked, which shows that document, in addition to information about the document topic proportions. Figure 10 shows the topic-document relation of the data set with 11 topics.

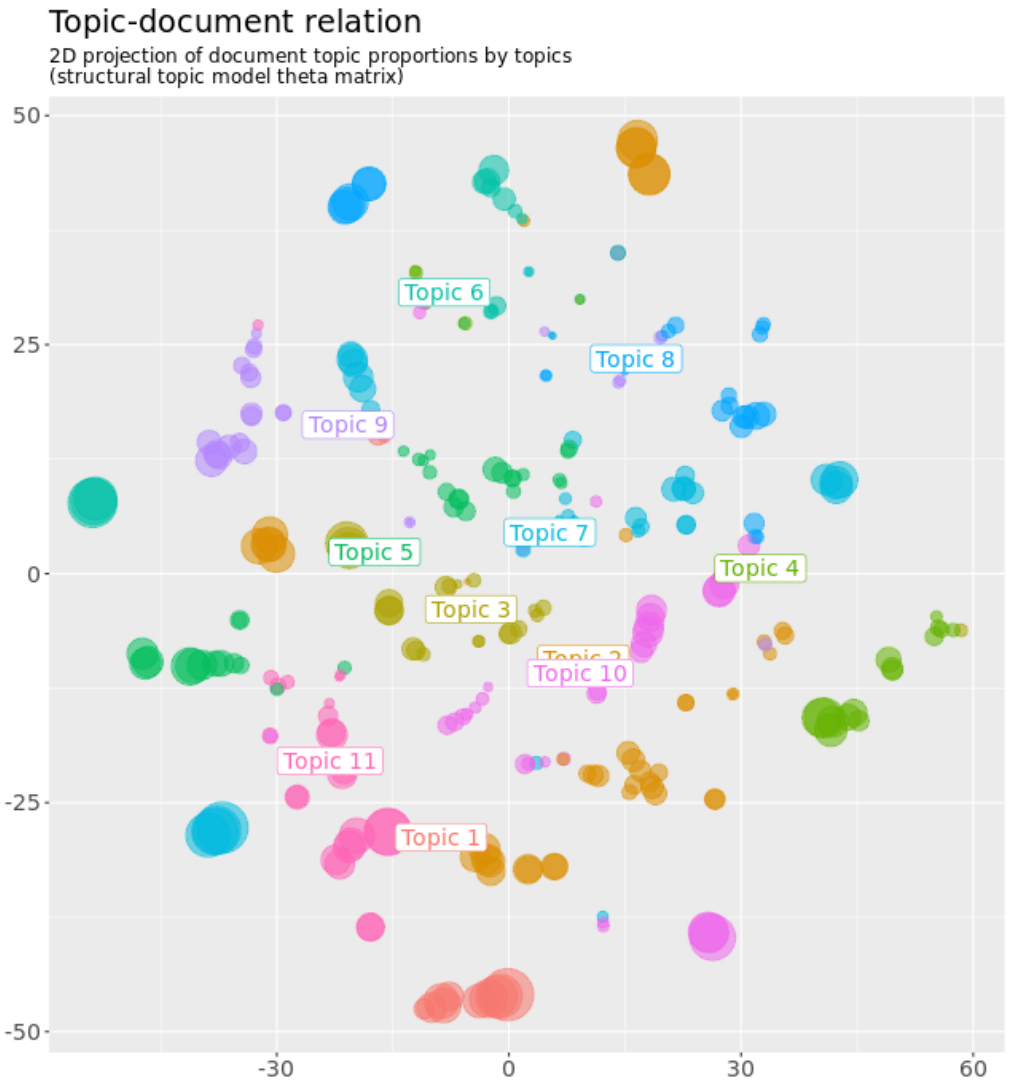

Figure 10. Topic-document relation

Figure 10 does not show clear clustering for the topics, and it seems like most of the topics are comprised of multiple clusters of documents. This can indicate that the topic count is too low, or it is simple a side effect of the documents not being internally cohesive and instead containing multiple different topics, which is allowed in STM. With higher topic counts the clusters become much clearer as can be seen in Figure 11 which shows the same data set with 25 topics.

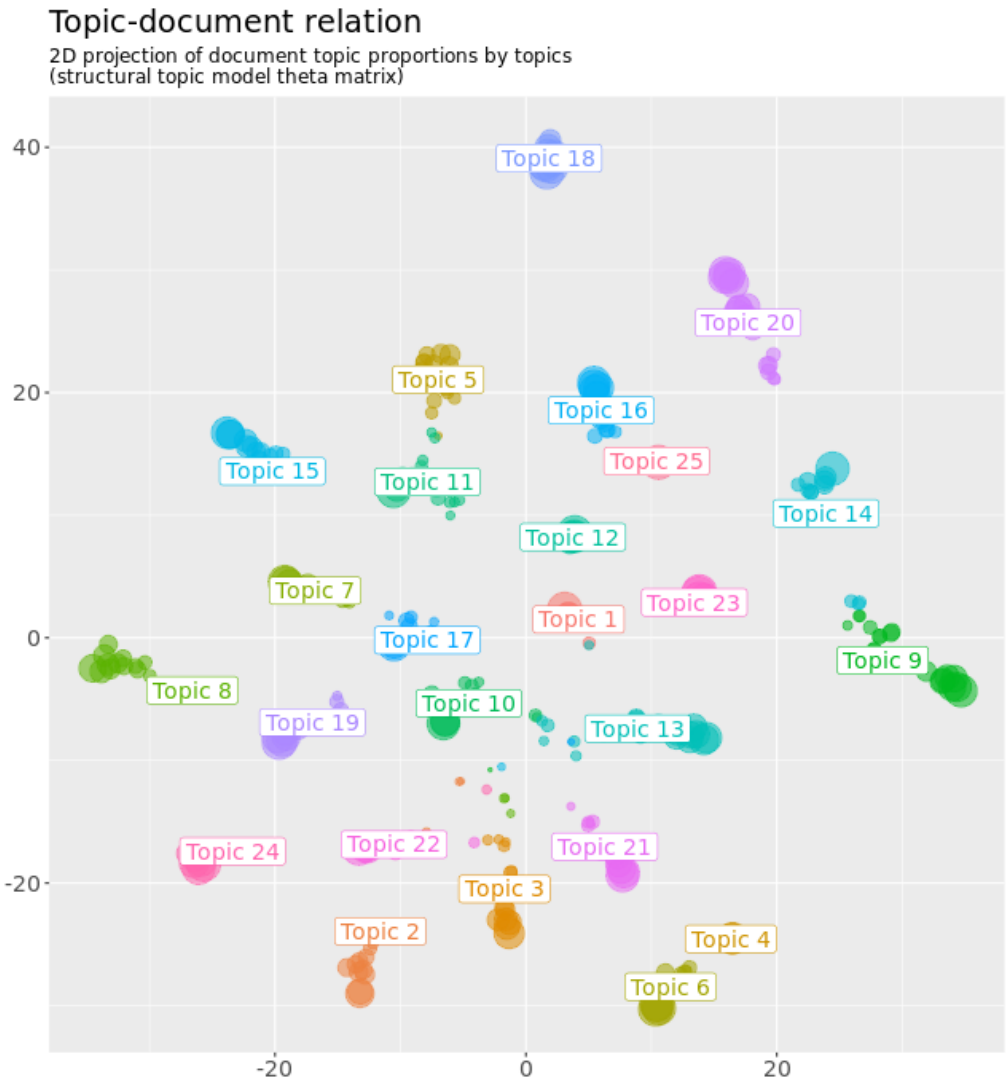

Figure 11. Topic-document relation with 25 topics

An example of similar documents clustering together can be seen in Figure 11 in the small banana-shape under the topic 20 label, where most of the documents are about not having enough teaching assistants in the exercise classes. Similar effect can be seen in the topic 8 cluster with documents mostly regarding the exercises being too much work. Figure 12 shows two examples from the banana-shape under topic 20. The document on the left translated in English “Too little help in the exercise classes”, and document on the right in English “Too few teaching assistants in the exercise classes.. Couldn’t receive help!”. Semantically the sentences are similar, and the meaning of the documents is mostly the same.

Document 232

Topic 20 29%, **Topic 14** 25%, **Topic 8** 8%, **Topic 9** 7%, **Topic 13** 4%, **Topic 11** 4%, **Topic 3** 3%, **Topic 5** 3%, **Topic 17** 2%, **Topic 2** 2%, **Topic 22** 2%, **Topic 15** 2%, **Topic 16** 2%, **Topic 7** 2%, **Topic 12** 1%, **Topic 25** 1%, **Topic 23** 1%, **Topic 19** 1%, **Topic 18** 1%,

Text:
Liian vähän harjoituksissa apua

Document 166

Topic 20 44%, **Topic 14** 17%, **Topic 8** 5%, **Topic 13** 5%, **Topic 16** 4%, **Topic 7** 4%, **Topic 11** 4%, **Topic 9** 3%, **Topic 3** 2%, **Topic 18** 2%, **Topic 23** 1%, **Topic 10** 1%, **Topic 22** 1%, **Topic 2** 1%, **Topic 15** 1%, **Topic 17** 1%, **Topic 25** 1%, **Topic 1** 1%,

Text:
liian vähän harkka-assareita harkoissa.. ei saanut apua!

Figure 12. Two examples of topic-document relation documents

Emotion analysis results are shown as percentages in an ordered flipped barplot of the eight NRC emotions. This should give a quick overview of the overall data set emotions. Figure 13 shows the emotion summary for the example data set. Sentiment summary is done similarly to the emotions but with only two values: positive and negative. Sentiment summary can be seen in Figure 14.

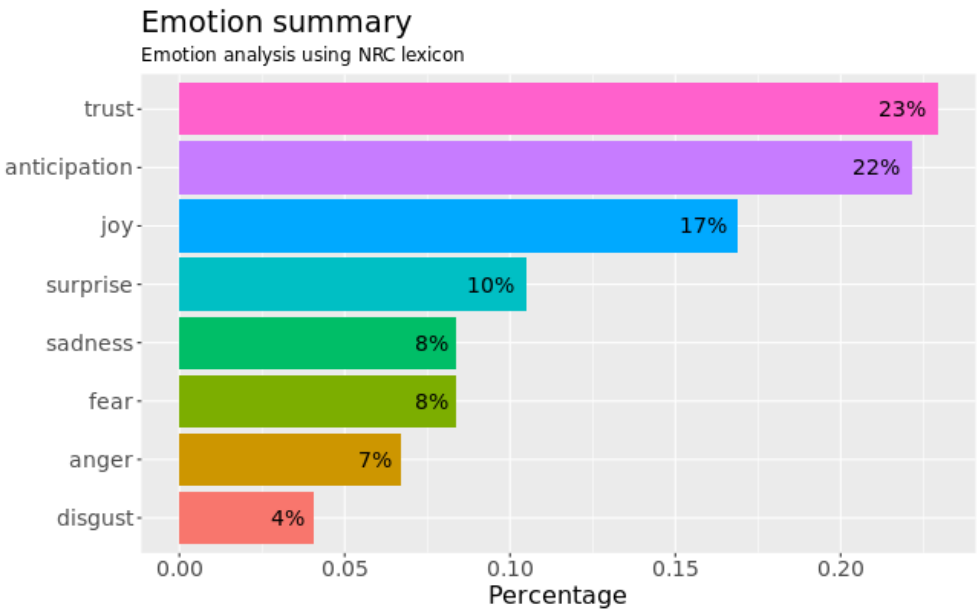

Figure 13. Emotion analysis summary

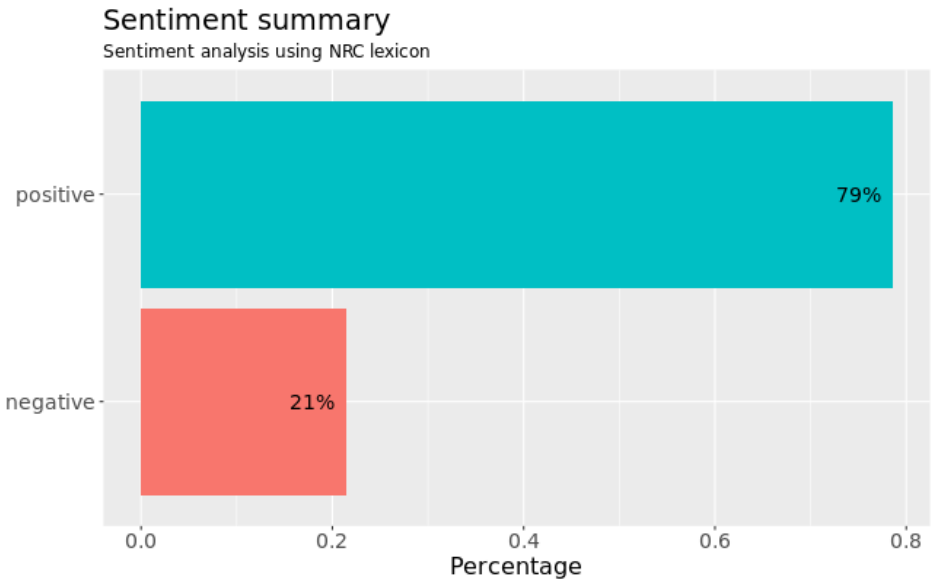

Figure 14. Sentiment analysis summary

All the plots in this section are contained in a summary page of the artefact. The goal of the summary page is to give an overview of the data analysis and the results. It should help judge the performance of the model and give the tools required to perform some exploratory data analysis.

5.4 Detailed topic information

The artefact contains a page with detailed information about each topic. An example of this can be seen in Figure 15 for the topic 9. A similar panel to Figure 15 is generated for each topic and the details page contains all these panels. User has the option to hide each of the smaller sections inside the panel using a filtering panel as shown in Figure 16. There are also options for sorting the emotion analysis results in descending or alphabetical order, as it can be easier to do comparisons between topics when the results are in the same order. Sentiment is shown as a single bar. Number of shown keywords and documents can be changed by the user. Keywords are selected in the same way as in the inter-topic distance plot, and the documents are selected in the order of highest topic prevalence. This information should aid the user understand what the topic is about by its vocabulary and example documents. The sentiment and emotions give additional insights about how, in this case, the survey

respondents feel about the specific topic. For example, if the examination was too hard in the course, and it is a recurring theme in the survey answers, it should end up as a topic that is negative and has a vocabulary that uses emotionally negative words.

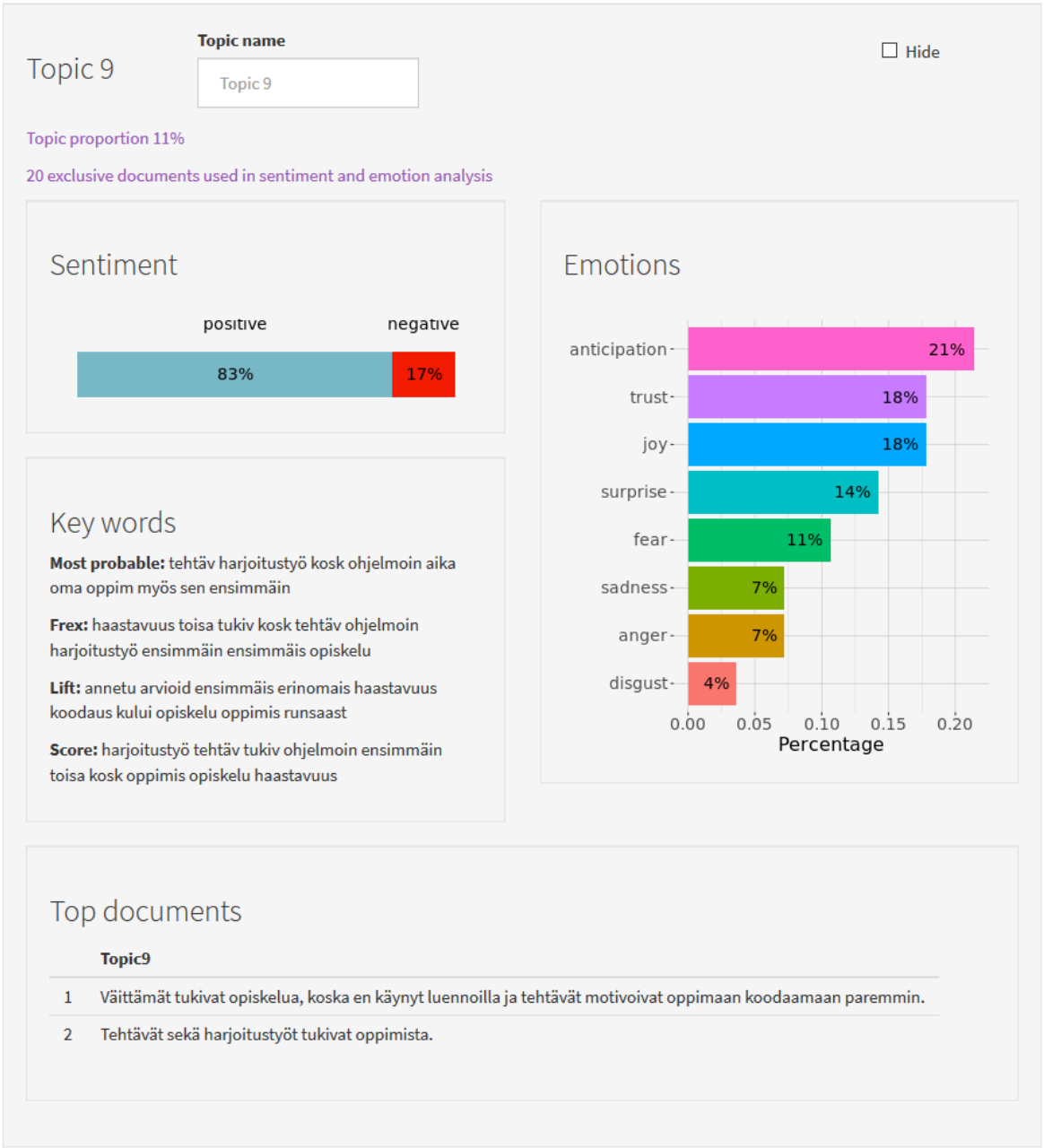

Figure 15. Detailed information of topic 9

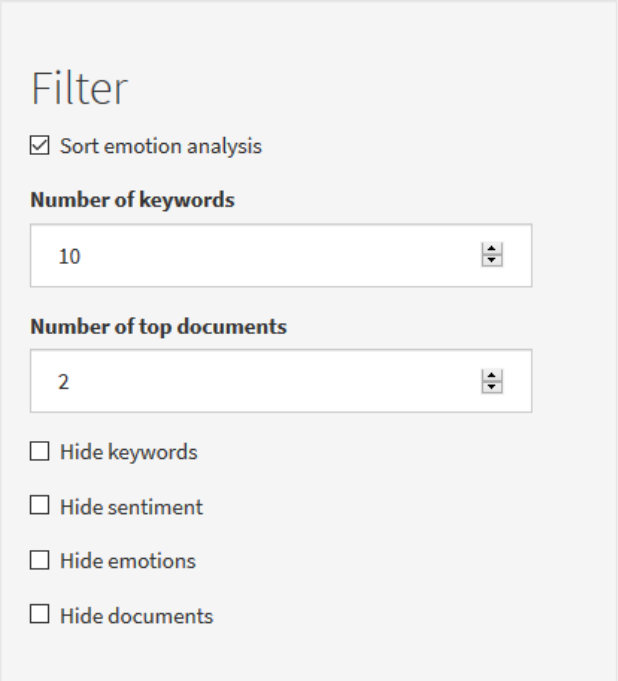

Figure 16. Topic details filter

5.5 Additional features and future improvements

There is a help page that shortly describes the application, how it is used and contains links to the most relevant used methods, this thesis and the source code. As the artefact is somewhat complicated, a help section should prove useful.

The artefact could be further improved by including covariate plots that show how different covariates affect the topics. This is an important advantage of the STM compared to other topic models, but it is not fully realized in the artefact.

It would be also possible to add quantitative analysis of the data, in addition to the qualitative analysis. It could aid in the selection of the right covariates from the data if the variances and mean values are shown to the user.

There are currently situations where the artefact silently fixes erroneous user inputs and does not inform the user. For example, it is possible to choose multiple content covariates from the data, but only the first one is used in the actual analysis. This can be misleading as user

is not given any warning of this happening. These should be fixed so that the user is always informed about changes to their selected options.

6 ARTEFACT EVALUATION

The artefact is evaluated using two methods. First, I demonstrate the usefulness and functionality of the artefact by performing analysis on a novice programming course feedback from 2019. Then, two researchers use Palaute to perform analysis and answer a questionnaire about the usefulness of the artefact.

These evaluation methods follow the DSR evaluation methods by (Sonnenberg and vom Brocke, 2012). More specifically ex post methods are used, meaning artefact is evaluated after its construction. The used methods are demonstration with the prototype and expert interview, although the interview is conducted via questionnaire.

Before the artefact is demonstrated or evaluated by experts, an evaluation of the performance of the stemmed Finnish NRC emotion lexicon is conducted. The issue of not finding matching words due to Finnish being inflected language arose during development of the artefact. It is tested if it makes sense to use stemmed version of the emotion lexicon to conduct emotion analysis. The initial testing showed a large increase in matched words, but it is unclear whether this increased the analysis accuracy or whether there are a lot of incorrectly identified words.

6.1 Stemmed lexicon performance evaluation

The performance of the stemmed lexicon was measured by coding the sentences manually and comparing the results with the coding received when using stemmed lexicon and the original NRC lexicon. In practice, this required a new data set. Novice programming course evaluation survey data from 2019 was used, which had a total of 121 responses. Combining the open answers to a single column makes the data set 300 documents. For easier analysis, the documents were further split down to individual sentences, resulting to a data set of 549 sentences, after which the emotion analysis was ran on the sentences. For each emotion and sentiment, top ten sentences with the highest correspondence were selected. So, ten sentences with the highest amount of identified anger words, ten sentences with most positive words and so forth totaling to 100 sentences, as there are eight NRC emotions and

two sentiments. After duplicates were removed there were 54 sentences left when using the original NRC lexicon. Stemmed lexicon was measured similarly, but the sentences were stemmed before analysis. There were 54 stemmed sentences left after duplicates were removed.

The results were saved as CSV-files that contain the sentences and the results as the rows, and each of the emotions and sentiments as columns. For later matching of the sentences, the sentence number was also saved in a column. Similar CSV-files were saved for the manual coding with the exception of all the analysis results being changed to zero. For the stemmed sentences, a column was included with the non-stemmed version of the sentence for manual coding. Finally, the order of the rows was randomized before the files were saved.

I manually coded both data sets. Coding was done on a binary scale of 0 or 1. The goal was to recognize if the sentence had particular emotions and sentiments in it or not. So, if for example, the sentence indicated joy, joy was marked 1 and if sadness was not indicated it was left 0. Every sentence could indicate multiple emotions, and this was often the case. This differs from the NRC results that show the number of identified words per emotion. It is not an intuitive way for humans to understand the sentiment and emotions by counting the words that seem to indicate some emotion, so coding was not done in the same way as the machine analysis does it. Instead, data was analyzed on the sentence level, as in what emotions are implied in this sentence.

Comparing the results of machine analysis to manual coding were done using a similarity calculation that accounted for the disparity between the coding methods. When machine analysis and manual coding both agree on an emotion, it was counted as a positive case. This includes machine analysis and manual coding both selecting zero, or both selecting anything other than zero. Negative cases include the other selecting zero and the other selecting anything but zero. Calculating the division of positive cases by all cases gives the similarity score of the sentence as a percentage. The similarities were calculated for every sentence and the results were saved to a vector of similarity scores with length equal to the number of the sentences. Calculating the mean of this similarity vector of all sentences gives the

similarity score between two data sets, which, in other words, is the accuracy of the machine analysis.

The results of comparing machine analysis with manual coding using similarity scores are in Table 3. Data sets original NRC and stemmed NRC were created using the machine analysis. Original manual and stemmed manual refer to the manually coded data sets. Random data sets were created by randomly assigning 0 and 1 as the emotion and sentiment values to the sentences. Random data sets were both compared 1000 times to the NRC data sets and the similarity score is calculated as the mean of those 1000 runs. As can be seen, both manually coded data sets perform better than randomly assigning values, although this difference is rather small when using the stemmed NRC lexicon.

Table 3. Comparison of lexicon performances

Data set A	**Data set B**	**Similarity**
Original NRC	Original manual	0.698
Original NRC	Original random	0.501
Stemmed NRC	Stemmed manual	0.567
Stemmed NRC	Stemmed random	0.500

A more detailed view of the different lexicon performances can be observed by looking at histograms. Figure 17 shows that the original NRC lexicon identifies emotions and sentiments mostly accurately and mostly above 50% with the largest accuracy around 60%, and none of the sentences are identified with less than 40% accuracy. This is different from Figure 18 that shows the histogram of the stemmed lexicon accuracy. Stemmed NRC lexicon has the most documents identified at lower a lower accuracy of around 50%. In addition to that, there is a large spike around 30% accuracy of sentences that are analyzed mostly incorrectly, while there are also some documents that are almost the opposite of the correct identification.

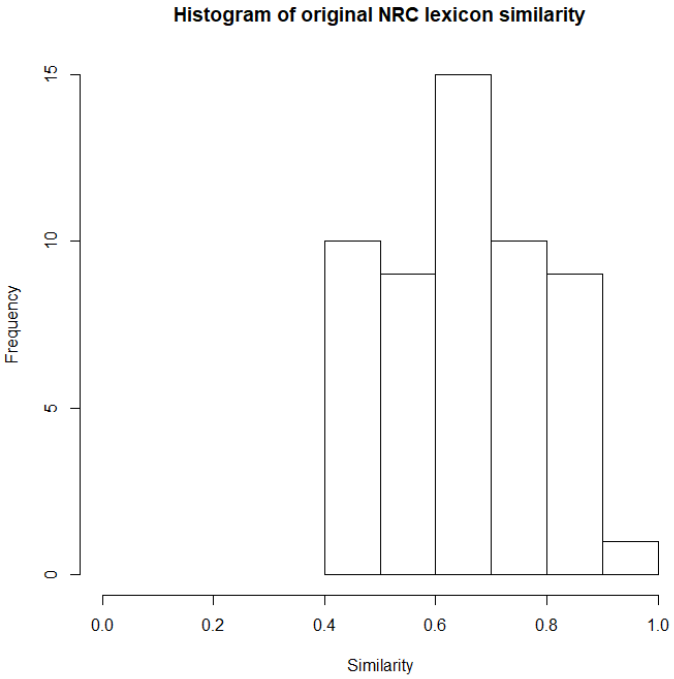

Figure 17. Histogram of original NRC lexicon similarity

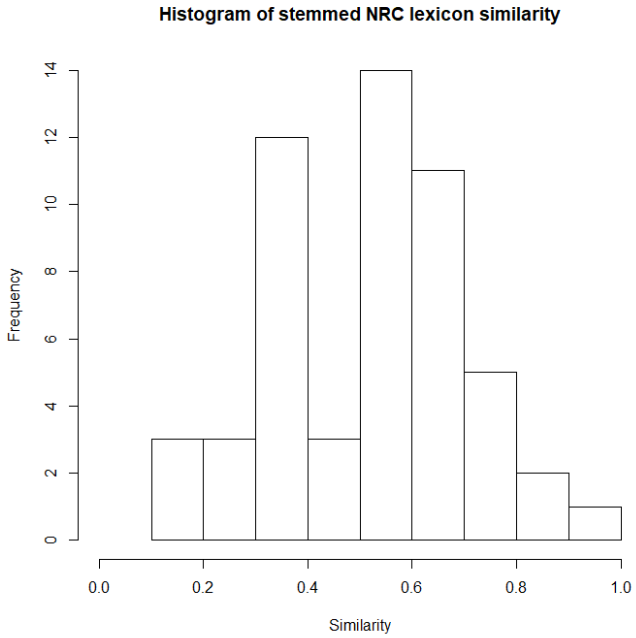

Figure 18. Histogram of stemmed NRC lexicon similarity

Qualitative analysis was performed on the most dissimilar sentences in the stemmed data set. All example sentences are in Finnish with English translation in parenthesis. One of the most incorrectly identified sentences was “Parasta tällä opintojaksolla omalla kohdalla oli tekemisen meininki ja onnistumisen riemu!” (“The best thing about this course for me was the good work attitude and the joy of success!”), which is clearly a wholly positive sentiment with positive emotions. The machine analysis still managed to identify every emotion and sentiment from that sentence, with anger being identified twice. A more careful look into the individual words in the sentence showed that the word “kohdalla” was stemmed as “kohd”, and this stem is identified having every emotion, when there really should not be any emotion or sentiment associated with this word. Another dissimilar sentence also had the word “kohdalla” and was thus identified incorrectly.

“Toisen periodin asiat alkoivat käydä itselle vaikeaksi ja kun en pystynyt panostamaan täysin asioihin, tipuin aika nopeasti kärryiltä.” (“The stuff in the second period began to get difficult for me, and when I couldn’t fully commit to the things, I quickly lost the plot.”) shows a negative sentiment and emotions of the course getting too difficult. It was incorrectly identified being very positive, trusting and full of anticipation. A closer look at the words showed that the word “kun” (“when”) was rated as having anticipation, joy, positive and trust associated with it. Similarly, “asia” (“thing”) was associated with anticipation, positive and trust, when it should be a completely neutral term.

Looking at more of the most wrongly identified sentences show similar results, where one or two stems are misidentified to have a lot of emotions and sentiments attached to them, when they should have none or just one. This is especially troublesome when it happens to common words like “on” (“is”), which is identified being full of positive emotions.

6.2 Demonstration on an introductory programming course

To demonstrate the functionality and capabilities of the artefact, course evaluation surveys are analyzed from a novice programming course from LUT university. The data set is the

same that was used to conduct the lexicon comparison, although the documents were not split down to individual sentences. So, a total of 121 responses to the survey with a total of 300 open answers. This analysis used the original NRC emotion lexicon as it was demonstrated to outperform the stemmed variant in the lexicon comparison.

The number of topics was hard to determine as the data set was very noisy. The documents varied in length greatly from single word documents to documents with word count around a hundred. As the documents were answers to open questions, they were written in the context of the question and it could make interpreting the answers more difficult when the question was not known anymore. This was also shown by some of the answers referring to other answers or questions. There were also miss-spelled words, compound words written together and separately and different characters like plus-signs, parenthesis, typed emoticons and numbers in the documents.

The sentiment analysis interprets the data as being very positive, with sentiment being 78% positive and 22% negative. Emotion analysis is overrun with trust, but fear is also relatively high suggesting that the data set is not just positive comments. The summary of emotions is shown in Figure 19.

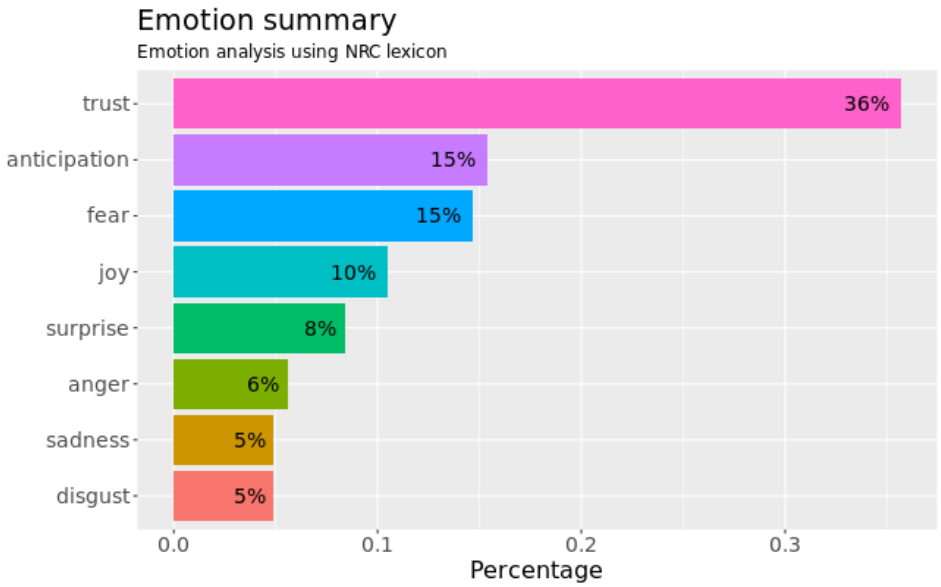

Figure 19. Novice programming course emotion analysis

Automatically determining the number of topics between 4 and 25 yielded 24 as the optimal number of topics. The automatic selection of topic number seems to prefer higher topic counts, possible due to more topics meaning more word exclusivity. 24 topics were hard to interpret and after much manual testing, 11 was selected as the best number of topics. The topics and their labels are shown in Table 4. The model converged with 11 topics after 347 iterations. A Likert-type question about the teaching methods supporting learning was used as a prevalence covariate.

Table 4. Programming course feedback topics

Topic	Topic proportion	Label
1	8%	Exercise classes were good
2	14%	Course materials and exercises were great
3	10%	Other
4	7%	Issues with the automatic code checker
5	9%	Course was well built
6	9%	Comments about exercises
7	12%	I learned a lot
8	8%	Heavy workload
9	8%	All the things that were good
10	8%	Exercise instructions were unclear
11	7%	Exam was too difficult

Going over the topics in Table 4, the theme in topic 1 was that the course exercise classes were deemed both useful and well implemented, with some distance participants wishing they would have been able to attend them. Some comments were hoping for more teaching assistants in the exercise classes, but they were in the minority.

In topic 2 the course and its materials were praised, in addition with the exercises. Topic 5 continues this trend by specifically praising the different teaching methods and technical aspects of the course like lectures, video lectures, course platform and the pacing of the course. Topic 9 is identified by the use of the word “good” as it is thrown around in multiple contexts. For example, the online implementation of the course was deemed good. Overall, the course received a lot of positive feedback.

Contrary to topic 2, topic 6 focused mainly about the negative sides of the exercises. Some participants liked the exercises while others had complaints about them being too much work and not useful. There is also topic 4 that focuses specifically about the issues of using automated checker for the exercises and how it was hard to make the checker accept the solution. So, the exercises were somewhat divisive.

Topic 7 has a clear focus on the learning aspect, with most of the documents stating that the respondent learned a lot. Topic 3 was the most difficult to interpret, but part of it is similar to topic 7 with the respondents stating to have learned during the course. Topic 3 deals with numerous aspects of the course with some stating it taught them the basics of the programming as is intended in a novice programming course, and others stating the workload was too much. There were also some ideas for improving the course.

Most of the topics contain individual documents that mention the heavy workload of the course. Topic 8 is mostly focused around that aspect of heavy workload. There are also multiple mentions that the lectures should be two hours instead of just one. The heavy workload comes from the exercises and the larger practical assignment at the end of the course according to the respondents.

Topic 10 focuses on the instructions that were unclear. This feedback was mostly weighted around the exam and the practical assignment having hard to follow instructions, but the normal exercises were also mentioned few times. In topic 11 the exam was also deemed too difficult compared to the other exercises in the course. Both topics are dominated by fear in

the emotion analysis, while topic 10 also has a high amount of the other negative emotions (anger, sadness, disgust).

Overall, the novice programming course was successful and received a lot of positive feedback. The course materials and methods seem to be good and do not need to be updated. The issues are the heavy workload and unclear instructions, especially in the exam. There have been issues with unfair grading in the past with this course, but this was not present anymore in the feedback. The workload is likely rated heavy since the course is mandatory for most of the students in LUT university whether they have programming experience or not and whether they are interested in programming or not. One way to fix the heavy workload would be to split the course into two separate introductory programming courses, where only the first part would be mandatory to most students and the second more advanced course is mandatory only to those studying programming. The exam instructions should be revised, as it was clearly causing issues for several students.

6.3 Expert review

Two experts were asked to use the artefact to analyze student course evaluation surveys and answer a questionnaire about the artefact afterwards. This method is like the expert interview from (Sonnenberg and vom Brocke, 2012), except it is conducted via questionnaires. The questionnaire was sent to the participants via email.

The questionnaire was designed following interview guidelines, as the purpose is more like an expert interview than a survey. Following the guidelines by deMarrais and Lapan (2003), the questionnaire questions were created to with a goal in mind. This goal was to answer the research questions 1 and 2 about the usefulness of the artefact and to improve the artefact. The participants were selected based on the understanding on topic modeling and emotion analysis. The questions were also reviewed by an expert, as is recommended (Marsden and Wright, 2010). The questions used in the questionnaire are shown in Table 5 along with the rationale. All the questionnaire questions are open.

Table 5. Reviewer questionnaire questions

Theme	**Question**
Background	1. What is your name?
	2. How familiar are you with topic modeling?
	3. How familiar are you with structural topic model (STM) and emotion analysis?
	4. What kind of data set did you analyze and were you familiar with the data set before analyzing it with Palaute?
RQ1 "Can the tool be used to analyze the intended data in a meaningful way?"	5. What kind of understanding of the data did you get using Palaute?
	6. Were the results obtained using Palaute useful in understanding the data? Why or why not?
	7. Do you think the results generated by Palaute are representative of the data set? Why or why not?
RQ2 "Does the intended user group deem the artefact useful?"	8. Do you think Palaute is useful in analyzing course evaluation survey answers? Why or why not?
	9. Do you think it is useful to read the survey answers after analyzing them with Palaute? Why or why not?
Improving the artefact	10. Were there any options or features missing that you would like to see implemented in Palaute?
	11. Was something unclear when using Palaute?
Additional comments	12. Additional thoughts or comments. For example, about the performance, UI or this questionnaire.

Going through the results, both participants were familiar with topic modeling, although one of them was not particularly familiar with STM. First participant used a larger data set from multiple courses, while the second used data from an English course.

Both participants agreed that Palaute helped them to understand the structure of the data and that the results were useful, although how it helped them understand the data varied a bit. For the first participant, the tool made it easy to understand the emotions and sentiment from the data, whereas for the second participant the tool streamlined and sped up the analysis process that they would usually do using different methods. Both participants also agreed that based on the literature, the tool gives representative results of the data set.

As for the usefulness of the artefact, both participants agreed that the graphical user interface makes it easy to use and thus makes it useful. As whether the student feedback should be still read manually after the analysis depends on the use-case. If you are a course teacher, then all feedback should still be read manually, but if the tool is used for analyzing multiple courses to get the bigger picture, then it is not necessary to manually read all the documents.

There were multiple improvements suggested by the participants. First, a summary that shows all the topics with their most important keywords would be useful, as currently you can see the topics only individually by clicking through the topic distance map (example in Figure 8). One of the participants missed the toggle for the default options and suggested adding functionality for selecting the number of topics. The toggle should be made more visible or moved closer to the start analysis button to make it clearer that you can select to not use the default options. There were also requests for downloading different parts of the analysis data as a CSV-file. This would make it easier to further conduct analysis on the data that is not limited by the functionality of Palaute.

Palaute had some parts that were deemed unclear. It was unclear what are the prevalence and content covariates and how they should be used. It was also unclear why some topics did not have a sentiment value. The covariates and how they should be used were explained in the help page of Palaute, but the help could also be included in the remap section of Palaute, as there the user is presented with the covariates. The sentiment is not identified when there are zero matches for the sentiment words, and this is indicated by the message "Sentiment was not identified". This message should be changed to better communicate why

the sentiment identification failed. Similar treatment should be done to the emotion analysis when it fails to find any emotions.

7 DISCUSSION

Looking at the lexicon comparison, the results in Table 3 suggest that using the original NRC gives more accurate results than using the stemmed NRC lexicon. Stemming the lexicon likely gives too many false matches, as words with different sentiment and emotion values are shortened to the same stem. While stemming the lexicon yields significantly more matched words, the matches tend to be incorrectly coded. Original NRC fails to match large number of words, but the matched words tend to be correct. A closer look at the most misidentified sentences shows that the incorrect identification is mostly a result of a few incorrect stems that wildly change the outcome of the analysis.

It is still unclear if the analysis of the complete data set is better with the stemmed NRC lexicon, as only 54 sentences of the 549-sentence corpus were manually analyzed. Although, if the sentences with most identified emotions have a lot of errors, the trend likely continues with the sentences with less identified emotions. It is thus better to use original NRC lexicon and not the stemmed variant. The stemmed NRC lexicon would need to be cleaned manually to improve its performance. The original NRC lexicon performance could be improved by lemmatizing the documents before the emotion analysis, since most of the words in the lexicon are already in their basic form. This should increase the number of matched words without introducing false matches.

There is a possibility of bias as I created the artefact and manually coded the data. I am aware of how the machine emotion analysis is performed, and this could influence how I manually coded the results. For less biased results, multiple coders should be used.

Some issues arose during the demonstration of Palaute with the novice programming course. First, it would make sense to allow using the question number as a covariate, since different questions are likely answered differently, but it is not implemented as of now. Currently in the details page, as is shown in Figure 15, the keyword likelihoods are not shown to the user. They are in the order of relevance, but without seeing the word likelihoods of belonging to a topic, it is hard to evaluate the importance of the word to the topic. So, showing ten

keywords might mean they are all very relevant, or only the first two keywords are relevant. There is a similar issue with the documents that are shown at the bottom of the topic detail panel. Without knowing the topic proportions of the documents, it is hard to evaluate what parts of the document are relevant to the topic. One interesting idea would be to color the words of the documents based on the word likelihoods. This would visually show what parts of the document are most relevant to the topic aiding in the interpretation process.

The comments from the experts indicate that Palaute was overall found to be useful in the task it was created for. It makes it easy to understand the structure of the data and performs sentiment and emotions analysis which is hard to do by hand. The experts also gave useful suggestions that would improve the tool further.

Palaute uses the inter-topic visualization from LDAvis by (Sievert and Shirley, 2014), but expands it by color-coding the topics by their sentiment. This novel addition is especially useful in text mining of reviews and other opinionated texts. Another change is using the visualization with STM instead of LDA, as STM is shown to outperform LDA (Roberts et al., 2014). LDAvis is still great for exploratory analysis, as the word-topic relations are visualized in a very interactive and informative way.

Palaute is an online service, meaning no external software needs to be installed. In this sense, it is similar to Sentiment Viz by (Healey and Ramaswany, 2019). Two main differences are that Palaute uses topic modeling in addition to sentiment and emotion analysis and Palaute works with all text data sets without being limited just to tweets. Although, this comes with the cost of the user having to input the data to Palaute.

Overall, combining topic modeling with sentiment and emotion analysis seems to be a novel idea, that is not widely explored in the literature already. Especially determining the topic specific sentiments and emotions has not been studied extensively.

8 CONCLUSION

Topic modeling and emotion analysis can be used in the educational context as a way of creating summaries of the data. Palaute is a tool that was created to accomplish that task. While it was originally intended as a tool for the teachers, it became clear during the development that it is more of an expert tool, since knowledge of the used technologies is required to operate the tool efficiently.

The goal for this thesis was to create an artefact that can be used to analyze course feedback in LUT university and to evaluate the analysis performance. The research questions specify this goal. To answer the first research question about using the tool to analyze the data in a meaningful way, based on the demonstration and evaluation of experts it seems like the tool manages to accomplish this task. Palaute helps in understanding the data, creating a summary of the data and creating information that would be hard to get from the data manually.

The main benefit of Palaute is the user interface that it provides to the complicated methods that are used under the hood. Performing topic modeling, emotion analysis and visualizing the results is not trivial, so automating this process is useful. The second research question deals with the usefulness of the artefact. The experts that reviewed Palaute both agreed that the artefact is useful. More specifically, Palaute is useful in understanding the structure of the data, and the graphical user interface makes the whole process of analyzing the data much easier than having to write the code for the analysis.

Last research question is about the accuracy of the emotion analysis. Based on the evaluation of the emotion analysis performance, Palaute gets the identified emotions mostly right at 69.8% accuracy. It is not clear how well the analysis is performed overall, as the impact of the documents with low amount of identified emotions was not measured. A test should be conducted with a manual coding of the whole data set to validate the emotion analysis performance further.

Palaute could be developed further. There are minor improvements that could be implemented like showing the topic keyword probabilities and topic document proportions. The analysis results should also be downloadable. Usability could be improved by always informing the user when some settings are incorrect and making it clearer when the user is using the default values. Some messages should also be worded better. There are larger features that could be added to overall improve the artefact like using columns as covariates or coloring the document words based on the probabilities of the topic word likelihoods.

9 REFERENCES

Agaoglu, M., 2016. Predicting Instructor Performance Using Data Mining Techniques in Higher Education. IEEE Access 4, 2379–2387. https://doi.org/10.1109/ACCESS.2016.2568756

Ahmad, S., Gupta, A., Gupta, N.K., 2019. Automated Evaluation of Students' Feedbacks using Text Mining Methods. IJRTE 8, 337–342. https://doi.org/10.35940/ijrte.D6846.118419

Ahonen, H., Heinonen, O., Klemettinen, M., Verkamo, A.I., 1997. Mining in the phrasal frontier, in: Komorowski, J., Zytkow, J. (Eds.), Principles of Data Mining and Knowledge Discovery. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 343–350. https://doi.org/10.1007/3-540-63223-9_133

Alhija, F.N.-A., Fresko, B., 2009. Student evaluation of instruction: What can be learned from students' written comments? Studies in Educational Evaluation 35, 37–44. https://doi.org/10.1016/j.stueduc.2009.01.002

Almjawel, A., Bayoumi, S., Alshehri, D., Alzahrani, S., Alotaibi, M., 2019. Sentiment Analysis and Visualization of Amazon Books' Reviews, in: 2019 2nd International Conference on Computer Applications Information Security (ICCAIS). Presented at the 2019 2nd International Conference on Computer Applications Information Security (ICCAIS), pp. 1–6. https://doi.org/10.1109/CAIS.2019.8769589

Biemann, C., Riedl, M., 2013. Text: now in 2D! A framework for lexical expansion with contextual similarity. JLM 1, 55. https://doi.org/10.15398/jlm.v1i1.60

Biggers, L.R., 2012. The effects of identifier retention and stop word removal on a latent Dirichlet allocation based feature location technique, in: Proceedings of the 50th Annual Southeast Regional Conference on - ACM-SE '12. Presented at the the 50th Annual Southeast Regional Conference, ACM Press, Tuscaloosa, Alabama, p. 164. https://doi.org/10.1145/2184512.2184551

Blei, D.M., 2012. Probabilistic topic models. Commun. ACM 55, 77. https://doi.org/10.1145/2133806.2133826

Blei, D.M., Lafferty, J.D., 2006. Correlated topic models, in: In Machine Learning: Proceedings of the Twenty-Third International Conference (ICML.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation 993–1022.

Buchanan, R., 1992. Wicked Problems in Design Thinking. Design Issues 8, 5. https://doi.org/10.2307/1511637

Caballero, A., Niguidula, J.D., Caballero, J.M., 2018. Analysis and Visualization of University Twitter Feeds Sentiment, in: Jung, J.J., Kim, P., Choi, K.N. (Eds.), Big Data Technologies and Applications. Springer International Publishing, Cham, pp. 132–145. https://doi.org/10.1007/978-3-319-98752-1_15

Chen, X., Vorvoreanu, M., Madhavan, K., 2014. Mining Social Media Data for Understanding Students' Learning Experiences. IEEE Transactions on Learning Technologies 7, 246–259. https://doi.org/10.1109/TLT.2013.2296520

Chowdhury, G.G., 2003. Natural Language Processing 39.

Correll, M., Li, M., Kindlmann, G., Scheidegger, C., 2019. Looks Good To Me: Visualizations As Sanity Checks. IEEE Transactions on Visualization and Computer Graphics 25, 830–839. https://doi.org/10.1109/TVCG.2018.2864907

Craft, B., Cairns, P., 2005. Beyond guidelines: what can we learn from the visual information seeking mantra?, in: Ninth International Conference on Information Visualisation

(IV'05). Presented at the Ninth International Conference on Information Visualisation (IV'05), pp. 110–118. https://doi.org/10.1109/IV.2005.28
Curiskis, S.A., Drake, B., Osborn, T.R., Kennedy, P.J., 2020. An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit. Information Processing & Management 57, 102034. https://doi.org/10.1016/j.ipm.2019.04.002
Da Silva Franco, R.Y., Abreu De Freitas, A., Santos Do Amor Divino Lima, R., Pereira Mota, M., Resque Dos Santos, C.G., Serique Meiguins, B., 2019. UXmood - A Tool to Investigate the User Experience (UX) Based on Multimodal Sentiment Analysis and Information Visualization (InfoVis), in: 2019 23rd International Conference Information Visualisation (IV). Presented at the 2019 23rd International Conference Information Visualisation (IV), pp. 175–180. https://doi.org/10.1109/IV.2019.00038
de Paula Santos, F., Lechugo, C.P., Silveira-Mackenzie, I.F., 2016. "Speak well" or "complain" about your teacher: A contribution of education data mining in the evaluation of teaching practices, in: 2016 International Symposium on Computers in Education (SIIE). Presented at the 2016 International Symposium on Computers in Education (SIIE), pp. 1–4. https://doi.org/10.1109/SIIE.2016.7751829
deMarrais, K.B., Lapan, S.D., 2003. Qualitative interview studies: Learning through experience, in: Foundations for Research. Routledge, pp. 67–84.
Eisenstein, J., Ahmed, A., Xing, E.P., Eisenstein, J., Ahmed, A., Xing, E.P., 2011. Sparse additive generative models of text, in: In Proc. ICML.
Eler, D.M., Pola, I.R.V., Garcia, R.E., Teixeira, J.B.M., 2018. Visualizing the Document Pre-processing Effects in Text Mining Process, in: Latifi, S. (Ed.), Information Technology - New Generations. Springer International Publishing, Cham, pp. 485–491.
Engelke, U., Abdul-Rahman, A., Chen, M., 2018. VISupply : A Supply-Chain Process Model fir Visualization Guidelines.
Farrell, R., Hooker, C., 2014. Values and Norms Between Design and Science. Design Issues 30, 29–38. https://doi.org/10.1162/DESI_a_00276
Farrell, R., Hooker, C., 2013. Design, science and wicked problems. Design Studies 34, 681–705. https://doi.org/10.1016/j.destud.2013.05.001
Feldman, R., Regev, Y., Hurvitz, E., Finkelstein-Landau, M., 2003. Mining the biomedical literature using semantic analysis and natural language processing techniques. BIOSILICO 1, 69–80. https://doi.org/10.1016/S1478-5382(03)02330-8
Fokkens, A., van Erp, M., Postma, M., Pedersen, T., Vossen, P., Freire, N., 2013. Offspring from Reproduction Problems: What Replication Failure Teaches Us, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Presented at the ACL 2013, Association for Computational Linguistics, Sofia, Bulgaria, pp. 1691–1701.
Garg, R., Heena, 2011. Study of text based mining, in: Proceedings of the International Conference on Advances in Computing and Artificial Intelligence - ACAI '11. Presented at the the International Conference, ACM Press, Rajpura/Punjab, India, pp. 5–8. https://doi.org/10.1145/2007052.2007054
Gottipati, S., Shankararaman, V., Lin, J.R., 2018. Text analytics approach to extract course improvement suggestions from students' feedback. RPTEL 13, 6. https://doi.org/10.1186/s41039-018-0073-0

Grönberg, N., 2020. Nikug/Palaute: Palaute. Zenodo. https://doi.org/10.5281/zenodo.3826075
Hashimi, H., Hafez, A., Mathkour, H., 2015. Selection criteria for text mining approaches. Computers in Human Behavior 51, 729–733. https://doi.org/10.1016/j.chb.2014.10.062
Healey, C., Ramaswany, 2019. Twitter Sentiment Visualization [WWW Document]. URL https://www.csc2.ncsu.edu/faculty/healey/tweet_viz/ (accessed 2.3.20).
Hevner, A.R., Chatterjee, S., 2010. Design research in information systems: theory and practice, Integrated series in information systems. Springer, New York ; London.
Hevner, A.R., March, S.T., Park, J., Ram, S., 2004. Design Science in Information Systems Research 32.
Hindle, A., Ernst, N.A., Godfrey, M.W., Mylopoulos, J., 2013. Automated topic naming: Supporting cross-project analysis of software maintenance activities. Empir Software Eng 18, 1125–1155. https://doi.org/10.1007/s10664-012-9209-9
Hu, M., Liu, B., 2004. Mining and Summarizing Customer Reviews.
Hu, N., Zhang, T., Gao, B., Bose, I., 2019. What do hotel customers complain about? Text analysis using structural topic model. Tourism Management 72, 417–426. https://doi.org/10.1016/j.tourman.2019.01.002
Hu, R., Rui, L., Zeng, P., Chen, L., Fan, X., 2018. Text Sentiment Analysis: A Review, in: 2018 IEEE 4th International Conference on Computer and Communications (ICCC). Presented at the 2018 IEEE 4th International Conference on Computer and Communications (ICCC), pp. 2283–2288. https://doi.org/10.1109/CompComm.2018.8780909
Jockers, M.L., 2015. Syuzhet: Extract Sentiment and Plot Arcs from Text.
Jordan, D.W., 2011. Re-thinking Student Written Comments in Course Evaluations: Text Mining Unstructured Data for Program and Institutional Assessment (Dissertation). California State University, Stanislaus.
Kandogan, E., Lee, H., 2016. A Grounded Theory Study on the Language of Data Visualization Principles and Guidelines. Electronic Imaging 2016, 1–9. https://doi.org/10.2352/ISSN.2470-1173.2016.16.HVEI-132
Kelleher, C., Wagener, T., 2011. Ten guidelines for effective data visualization in scientific publications. Environmental Modelling & Software 26, 822–827. https://doi.org/10.1016/j.envsoft.2010.12.006
Kember, D., Leung, D.Y.P., Kwan, K.P., 2002. Does the Use of Student Feedback Questionnaires Improve the Overall Quality of Teaching? Assessment & Evaluation in Higher Education 27, 411–425. https://doi.org/10.1080/0260293022000009294
Kettunen, K., 2005. Developing an automatic linguistic truncation operator for best-match retrieval of Finnish in inflected word form text database indexes.
Kettunen, K., Baskaya, F., 2011. Stemming Finnish for Information Retrieval – Comparison of an Old and a New Rule-based Stemmer. Proceedings of the 5th Language & Technology Conference (LTC 2011), Poznan 5.
Khoo, C.S., Johnkhan, S.B., 2018. Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons. Journal of Information Science 44, 491–511. https://doi.org/10.1177/0165551517703514
Korenius, T., Laurikkala, J., Järvelin, K., Juhola, M., 2004. Stemming and lemmatization in the clustering of finnish text documents, in: Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management - CIKM '04. Presented at

the the Thirteenth ACM conference, ACM Press, Washington, D.C., USA, p. 625. https://doi.org/10.1145/1031171.1031285

Koto, F., Adriani, M., 2015. HBE: Hashtag-Based Emotion Lexicons for Twitter Sentiment Analysis, in: Proceedings of the 7th Forum for Information Retrieval Evaluation on - FIRE '15. Presented at the the 7th Forum for Information Retrieval Evaluation, ACM Press, Gandhinagar, India, pp. 31–34. https://doi.org/10.1145/2838706.2838718

Koufakou, A., Gosselin, J., Guo, D., 2016. Using data mining to extract knowledge from student evaluation comments in undergraduate courses, in: 2016 International Joint Conference on Neural Networks (IJCNN). Presented at the 2016 International Joint Conference on Neural Networks (IJCNN), pp. 3138–3142. https://doi.org/10.1109/IJCNN.2016.7727599

Kumar, A., Ekbal, A., Kawahra, D., Kurohashi, S., 2019. Emotion helps Sentiment: A Multi-task Model for Sentiment and Emotion Analysis, in: 2019 International Joint Conference on Neural Networks (IJCNN). Presented at the 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8852352

Lucas, C., Nielsen, R.A., Roberts, M.E., Stewart, B.M., Storer, A., Tingley, D., 2015. Computer-Assisted Text Analysis for Comparative Politics. Polit. anal. 23, 254–277. https://doi.org/10.1093/pan/mpu019

Maaten, L. van der, 2014. Accelerating t-SNE using Tree-Based Algorithms. Journal of Machine Learning Research 15, 3221–3245.

Marsden, P.V., Wright, J.D. (Eds.), 2010. Handbook of survey research, Second edition. ed. Emerald, Bingley, UK.

Marsh, H.W., 1984. Students' Evaluations of University Teaching : Dimensionality , Reliability , Validity , Potential Biases , and Utility. https://doi.org/10.1037/0022-0663.76.5.707

Mimno, D., McCallum, A., 2008. Topic Models Conditioned on Arbitrary Features with Dirichlet-multinomial Regression 8.

Mohammad, S., Salameh, M., Kiritchenko, S., 2016. Sentiment Lexicons for Arabic Social Media, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). Presented at the LREC 2016, European Language Resources Association (ELRA), Portorož, Slovenia, pp. 33–37.

Mohammad, S.M., Yang, T., 2013. Tracking Sentiment in Mail: How Genders Differ on Emotional Axes. arXiv:1309.6347 [cs].

Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S., 2007. A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems 24, 45–77. https://doi.org/10.2753/MIS0742-1222240302

Phan, X.-H., Nguyen, L.-M., Horiguchi, S., 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections, in: Proceeding of the 17th International Conference on World Wide Web - WWW '08. Presented at the Proceeding of the 17th international conference, ACM Press, Beijing, China, p. 91. https://doi.org/10.1145/1367497.1367510

Pong-Inwong, C., Kaewmak, K., 2016. Improved sentiment analysis for teaching evaluation using feature selection and voting ensemble learning integration, in: 2016 2nd IEEE International Conference on Computer and Communications (ICCC). Presented at

the 2016 2nd IEEE International Conference on Computer and Communications (ICCC), pp. 1222–1225. https://doi.org/10.1109/CompComm.2016.7924899
Porter, M.F., 2001. Snowball: A language for stemming algorithms 20.
Rittel, H.W.J., Webber, M.M., 1973. Dilemmas in a General Theory of Planning. Policy Sciences 4, 155–169.
Roberts, M.E., Stewart, B.M., Airoldi, E.M., 2016. A Model of Text for Experimentation in the Social Sciences. Journal of the American Statistical Association 111, 988–1003. https://doi.org/10.1080/01621459.2016.1141684
Roberts, M.E., Stewart, B.M., Tingley, D., 2019. stm: An R Package for Structural Topic Models. J. Stat. Soft. 91. https://doi.org/10.18637/jss.v091.i02
Roberts, M.E., Stewart, B.M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S.K., Albertson, B., Rand, D.G., 2014. Structural Topic Models for Open-Ended Survey Responses: STRUCTURAL TOPIC MODELS FOR SURVEY RESPONSES. American Journal of Political Science 58, 1064–1082. https://doi.org/10.1111/ajps.12103
Roberts, M.E., Tingley, D., Stewart, B.M., Airoldi, E.M., 2013. The Structural Topic Model and Applied Social Science. Advances in Neural Information Processing Systems Workshop on Topic Models: Computation, Application, and Evaluation 4.
Robinson, J.S. and D., n.d. 6 Topic modeling | Text Mining with R.
Romero, C., Ventura, S., 2007. Educational data mining: A survey from 1995 to 2005. Expert Systems with Applications 33, 135–146. https://doi.org/10.1016/j.eswa.2006.04.005
Russell, J.A., 1980. A circumplex model of affect. Journal of Personality and Social Psychology 39, 1161–1178. https://doi.org/10.1037/h0077714
Sanchez, D., Martin-Bautista, M.J., Blanco, I., de la Torre, C.J., 2008. Text Knowledge Mining: An Alternative to Text Data Mining, in: 2008 IEEE International Conference on Data Mining Workshops. Presented at the 2008 IEEE International Conference on Data Mining Workshops, IEEE, Pisa, pp. 664–672. https://doi.org/10.1109/ICDMW.2008.57
Shneiderman, B., 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations.
Sievert, C., Shirley, K., 2014. LDAvis: A method for visualizing and interpreting topics, in: Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces. Presented at the Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces, Association for Computational Linguistics, Baltimore, Maryland, USA, pp. 63–70. https://doi.org/10.3115/v1/W14-3110
Sliusarenko, T., Harder Clemmensen, L., Kjær Ersbøll, B., 2013. Text Mining in Students' Course Evaluations - Relationships between Open-ended Comments and Quantitative Scores:, in: Proceedings of the 5th International Conference on Computer Supported Education. Presented at the 5th International Conference on Computer Supported Education, SciTePress - Science and and Technology Publications, Aachen, Germany, pp. 564–573. https://doi.org/10.5220/0004384705640573
Sonnenberg, C., vom Brocke, J., 2012. Evaluation Patterns for Design Science Research Artefacts, in: Helfert, M., Donnellan, B. (Eds.), Practical Aspects of Design Science, Communications in Computer and Information Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 71–83. https://doi.org/10.1007/978-3-642-33681-2_7

Sousa Santos, B., Dias, P., 2013. Evaluation in visualization: some issues and best practices, in: Wong, P.C., Kao, D.L., Hao, M.C., Chen, C. (Eds.), . Presented at the IS&T/SPIE Electronic Imaging, San Francisco, California, USA, p. 90170O. https://doi.org/10.1117/12.2038259

Stojanovski, D., Strezoski, G., Madjarov, G., Dimitrovski, I., Chorbev, I., 2018. Deep neural network architecture for sentiment analysis and emotion identification of Twitter messages. Multimed Tools Appl 77, 32213–32242. https://doi.org/10.1007/s11042-018-6168-1

Stupans, I., McGuren, T., Babey, A.M., 2016. Student Evaluation of Teaching: A Study Exploring Student Rating Instrument Free-form Text Comments. Innov High Educ 41, 33–42. https://doi.org/10.1007/s10755-015-9328-5

Tabak, F.S., Evrim, V., 2016. Comparison of emotion lexicons, in: 2016 HONET-ICT. Presented at the 2016 HONET-ICT, pp. 154–158. https://doi.org/10.1109/HONET.2016.7753440

Tao, J., Fang, X., 2020. Toward multi-label sentiment analysis: a transfer learning based approach. J Big Data 7, 1. https://doi.org/10.1186/s40537-019-0278-0

Tedmori, S., Awajan, A., 2019. Sentiment Analysis Main Tasks and Applications: A Survey. JIPS 15, 500–519. https://doi.org/10.3745/JIPS.04.0120

Tseng, C.-W., Chou, J.-J., Tsai, Y.-C., 2018. Text Mining Analysis of Teaching Evaluation Questionnaires for the Selection of Outstanding Teaching Faculty Members. IEEE Access 6, 72870–72879. https://doi.org/10.1109/ACCESS.2018.2878478

Tucker, B., 2014. Student evaluation surveys: anonymous comments that offend or are unprofessional. High Educ 68, 347–358. https://doi.org/10.1007/s10734-014-9716-2

Vinten, G., 1995. Open versus closed questions – an open issue. Management Decision 33, 27–31. https://doi.org/10.1108/00251749510084653

Wang, W., Chen, L., Thirunarayan, K., Sheth, A.P., 2012. Harnessing "Big Data" for Automatic Emotion Identification, in: 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing. Presented at the 2012 International Conference on Privacy, Security, Risk and Trust (PASSAT), IEEE, Amsterdam, Netherlands, pp. 587–592. https://doi.org/10.1109/SocialCom-PASSAT.2012.119

Wang, X., Goh, D.H.-L., 2020. Components of game experience: An automatic text analysis of online reviews. Entertainment Computing 33, 100338. https://doi.org/10.1016/j.entcom.2019.100338

Zabaleta, F., 2007. The use and misuse of student evaluations of teaching. Teaching in Higher Education 12, 55–76. https://doi.org/10.1080/13562510601102131

Zhang, W., Qin, S., Jin, H., Deng, J., Wu, L., 2017. An Empirical Study on Student Evaluations of Teaching Based on Data Mining. EURASIA J. Math., Sci Tech. Ed 13, 5837–5845. https://doi.org/10.12973/eurasia.2017.01033a