FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Configurable Text Exploration Interface with NLP for Decision Support

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Medieninformatik und Visual Computing

eingereicht von

Martin Mattäus Śmiech
Matrikelnummer 01426853

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Dr.techn. Manuela Waldner, MSc

Wien, 22. März 2018

Martin Mattäus Śmiech	Manuela Waldner

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Configurable Text Exploration Interface with NLP for Decision Support

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Media Informatics and Visual Computing

by

Martin Mattäus Śmiech
Registration Number 01426853

to the Faculty of Informatics

at the TU Wien

Advisor: Dr.techn. Manuela Waldner, MSc

Vienna, 22nd March, 2018

Martin Mattäus Śmiech   Manuela Waldner

Erklärung zur Verfassung der Arbeit

Martin Mattäus Śmiech
Hermanngasse 2a/201

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 22. März 2018

Martin Mattäus Śmiech

Danksagung

Ich danke meiner Betreuerin Dr. techn. Manuela Waldner MSc für regelmäßige Unterstützung und Feedback. Dieses Projekt wurde in Zusammenarbeit mit PS Quant bzw. dessen Vertretern Michael Pühringer und Sebastian Schrey durchgeführt, welche fachspezifische Expertise und nützliche Informationen beigetragen haben.

Die Umsetzung des UIs/Frameworks wurde gemeinsam mit Dea Čizmić und ihrer Bachelorarbeit bewerkstelligt. Die globale Visualisierung für Topic Modeling auf Basis von NMF war angelehnt an die Implementierung von "BiSet edge bundeling"von Lukas Eibensteiner und Manuel Franz Josef Kapferer, ebenso wie an den Anpassungen daran von Michael Mazurek.

Acknowledgements

I want to express my gratitude to my advisor Dr. techn. Manuela Waldner MSc for continuous support, guidance and feedback. This work has been developed in cooperation with PS Quant whose representatives Michael Pühringer and Sebastian Schrey provided domain specific knowledge and useful materials during the development of this project.

The implementation of the UI/framework has been done in collaboration with Dea Čizmić and her bachelor thesis. The global visualization for topic modeling based on NMF has been derived from "BiSet edge bundeling" by Lukas Eibensteiner and Manuel Franz Josef Kapferer as well as adaptations made to it by Michael Mazurek.

Abstract

Having to read and understand lots of text documents and reports on a daily basis can be quite challenging. The intended audience for these reports has limited resources and wants to reduce time spent on reading such reports. Therefore a need for a tool emerges that assists the process of gaining relevant information out of reports/documents more quickly. These text documents are often unstructured and of varying length. They are written in the English language and are available from different sources (such as RSS feeds and text files). The aim of this project is to offer a tool that supports the process of analysing and understanding given texts. This is made possible by using natural language processing (NLP) and text visualization (TextVis). TextVis is already a well known and frequently used solution. The herein described project uses an NLP pipeline which serves as preprocessing for TextVis. To provide quick insight into the data, topic extraction mechanisms like Latent Dirichlet Allocation (LDA) or Non-negative Matrix Factorization (NMF) are available for the user to be chosen within the aforementioned pipeline. A major challenge for TextVis is the configuration of the NLP pipeline, because there are many different ways of doing so and a wide range of parameters to chose from. To overcome this issue, this project provides a solution that enables users to easily configure and customize their own NLP pipeline. It is designed to encourage these users to experiment with different sequences of NLP operations and parameter configurations to find a solution that suites them best. In order to keep it easy to use the software, it is implemented entirely using web technologies to be accessible in a common web browser. The resulting visualization will emphasize particular parts of the text based on a set of different factors, if selected so. These factors can be topics, sentiments and part-of-speech-tagged words. The focus of this work lies on a visual interface that enables and encourages users to adjust/optimize the underlying NLP pipeline (by selecting steps and setting parameters) and comparing their results. Evaluation with help of user feedback showed that certain pipeline configurations work better for certain types of texts than others. Using the solution created within this work, users can adapt the tool to their needs and also tweak it according to requirements. There is no universal configuration that works for all documents, however.

Contents

CHAPTER 1

Introduction

This work focuses on processing and visually representing text reports written in English. The stakeholders of this project are especially interested in the commodity market. Companies in this domain have to make decisions whether to buy or not to buy a certain kind of commodity based on such reports. The employees of these companies have to read dozens of lengthy reports every day. Their resources to do so are limited, hence the need for a tool that helps them to minimize the time spent reading such reports emerges.

Answering what a document is about and whether the situation is good or bad is crucial to the decision making process. Ideally, a solution would do that by automatically issuing recommendations to perform certain actions (like e.g. "buy nickel today") based on gained knowledge of given documents. This ideal solution cannot be accomplished trivially and is analytically impossible. To issue such recommendations sophisticated domain and context specific knowledge is necessary which cannot be depicted by the system. This is a classic visual analytics problem. On top of that, there is missing trust on the users side. To deal with this, the herein described project is designed to include the user into the process of analysis and evaluation of the reports/documents.

Therefore, the goal of this project is to approximate the aforementioned ideal solution by visually summarizing the key concepts of text documents. These documents are analysed using natural language processing (NLP) operations. This sequence of operations is called NLP pipeline. An NLP pipeline consists of multiple NLP operations. Most of these steps have parameters. However, there is no gold standard for how to construct an NLP pipeline. Hence, instead of predetermining a fixed pipeline configuration, the core of this solution is to let the user easily create a custom NLP pipeline. The end result is a visual summary of the processed text and applied pipeline. This aids the decision making process by offering insight specifically configured based on the users input.

This work contributes to the area of natural language processing. It does so by offering:

1. A configurable NLP pipeline with various visual outputs.
2. Integration into a financial dashboard (web based).
3. Tool assisted evaluation of different NLP pipelines.

CHAPTER 2

Related Work

On the one hand, this work uses NLP, on the other hand, NLP is preprocessing and its results need to be visualized. The ideal means for this task is text visualization (TextVis). There are related works, such as software tools, which are of varying specialization. They can be divided into different approaches, platforms and languages. Some of the available tools are only accessible to professionals to use and tweak because of the lack of a compelling (graphical) user interface. Others only provide a simple output with no possibility to interact with it.

2.1 Natural Language Processing

NLP is necessary to create visual encodings. It is a broad field, however. There are various known features and operations to it, the most relevant from this works perspective are the following: Tokenization, Part-of-Speech (PoS) Tagging, Named Entity Recognition (NER), Stemming, Lemmatization, Sentiment Analysis and Topic Modeling. A brief explanation of each of these features follows.

Tokenization takes care of segmenting or splitting an input document into smaller parts, called tokens [MSB^{+}14].

PoS Tagging estimates what grammatical tag a word or token has [MSB^{+}14]. That means that a word can be, for example, a noun or a verb. Furthermore, some PoS taggers can be used to determine whether the word is in plural or singular form.

NER detects whether a token is a named entity, which can be a place, date, organization, etc [MSB^{+}14].

Stemming takes a word in a certain form and tries to find a base or stem form of it, e.g. "walking" becomes "walk" [SG16].

Lemmatization serves to find different words with the same meaning [SG16].

Sentiment Analysis tries to determine the attitude or emotion of a text segment or document [MDP+11].

Topic Modeling creates a topic model that represents statistical distribution or weight of certain topics across documents [Wal06].

Some of these operations are usually performed at a certain order. This order can be used to define a sequence that can be referred to as NLP pipeline. Refer to Chapter 3 to see how such a pipeline works.

There are tools that are able to provide multiple techniques at once or in series like OpenNLP, CoreNLP, NLTK which offer numerous features and services [awe18]. The most prominent of these kinds of libraries are written in Python and some in Java. There are only a few tools that are actually available for use in the browser, which was a requirement for this thesis.

The Stanford Natural Language Processing Toolkit called CoreNLP features a broad range of NLP features to be used [MSB+14]. It is developed in Java and offers a common API for developers to perform NLP operations and the ability to create a pipeline according to their respective goals and needs. Among the included features are: tokenization, NER and sentiment analysis. CoreNLP has a server component simply called "CoreNLP Server", which provides a graphical user interface (GUI) for CoreNLP features via web [cor]. Within that GUI, it is possible to create an annotation pipeline which is comparable to this work's NLP pipeline configuration interface, further described in Section 3. Pipelines made within the CoreNLP Server GUI are still created using the Java backend, in contrast to this work, which has a pure web client-side implementation.

A similar work to CoreNLP is OpenNLP, which is also a Java based technology that provides NLP operations via an API [Ope]. OpenNLP supports all of the aforementioned NLP operations except sentiment analysis. The previously described related work Document Cards uses OpenNLP for most of its preprocessing steps [SOR+09]. In contrast to CoreNLP and the herein described project, OpenNLP does not offer a GUI and is targeted for developers. It also does not provide topic modeling but offers a more sophisticated approach to sentiment analysis.

A prominent example for NLP software tools is the Natural Language Toolkit (NLTK) [LRR11]. Functionality wise it offers similar features to CoreNLP and OpenNLP. Like OpenNLP, NLTK does not have a GUI but only a developer API. It provides a rich resource set including 60 corpora, grammar collections and trained models. It is written in Python which has advantages like high readability, an object oriented paradigm, easy extensibility and a powerful standard library. Python also supports read-eval-print loops (REPL) which make it possible to quickly evaluate expressions within a shell without having to set up a software project. NLTK is a widely used and strongly supported tool. Due to design constraints (entire processing in a web browser), NLTK was no option for this work. It would have required a dedicated server which runs Python and offers a web accessible NLTK API.

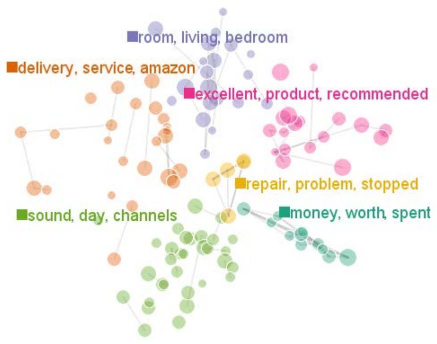


Figure 2.1: UTOPIAN Topic Modeling Visualization Result [CLRP13].

A different example is User-Driven Topic Based on Interactive Nonnegative Matrix Factorization (UTOPIAN), which is a work that focuses on technical and mathematical aspects of topic modeling [CLRP13]. UTOPIAN is relevant herein because it tries to provide helpful visual results, which can be seen in Figure 2.1. These results are based on text documents to support the user's decision making process. Natural language text documents serve as inputs and will be processed in order to create a topic-wise representation of them. All of this happens underneath a complex UI created in Java. It uses a semi-supervised NMF method to extract weighted combinations of clusters to showcase their proximity [CLRP13]. Compared to this work, UTOPIAN does not offer configuration of the underlying NLP pipeline. Users can influence the classification process of the topics/keywords, though. The presentation relies on global results (or visualizations) rather than multiple individual results for different data sources.

2.2 TextVis

Since this work essentially results in a text visualization of given input texts, the most relevant comparison is to other works in the field of text visualization (TextVis). In TextVis, there is a great range of different techniques, most of which usually provide user interaction to adjust the resulting visualization by influencing visual properties like color, size and distance, for example. There is a shear amount of different approaches, starting from text highlighting and font properties to complex multi dimensional visualizations. To deal with the variety of different visualization techniques, there is a project called Text Visualization Browser [KK15] dedicated to visually survey them.

One example is Guideline for Effective Usage of Text Highlighting Techniques [SOK^{+}16] which is of particular relevance herein since this project uses many of these techniques or similar ones . This work focuses on how to effectively and automatically emphasize text elements and how to create interactive visual interfaces to do so. Particular examples of proposed techniques are typesetting, background coloring, font weight and underlining.

To estimate what text segments are of interest and what features (statistical, syntactic, semantic and structural) they have, it is necessary to apply natural language processing. Three different studies have been conducted which led to the following results: font size, background color and border are the most effective approaches to emphasize text. Font spacing, italics and underlines should be avoided since they are perceived as distracting. This project makes use of recommendations issued in this related work regarding background color, font color and interference avoidance (conjunction of features). In contrast to the herein described project, this related work focuses on means of tweaking and adjusting visual properties based on text features rather than the way the text itself is being analysed.

A text visualization method for cross-domain research topic mining [JZ16] is a work that creates a visual representation of the change of different scientific topics across multiple domains over time. It offers a user interface that enables users to explore and experiment with the visualization to gain a better understanding of the relationship of the topics. It uses a combination of different plot types to present the data, including Sankey diagrams, scatter plots, and word clouds. This related work focuses on presenting the evolution of individual topics across domains over time, whereas this work focuses on the topics of multiple documents independent from time.

Another interesting example is Document Cards (DC) which tries to represent documents in compact sized cards and also uses NLP to do so [SOR+09]. Among the applied NLP operations are sentence splitting, part-of-speech tagging and noun phrase chunking. DC arranges multiple cards next to each other. A card is a rectangular shape that is arranged in a grid. Each of the cards has interactive properties that present further information when, for example, hovering with the cursor above it. To gather semantic information out of the documents it uses text and keyword extraction to summarize the documents. In contrast to this work, DC uses a corpus independent scoring method to rank the keywords. Furthermore, DC focuses on picture-like output and representation of the document. The cards are being treated like picture elements and text is not meant to be readable. It is designed to be used on small screen devices. DC is still of particular interest in comparison to the herein described project since it uses cards as well, just in a different manner.

CHAPTER 3

Natural Language Pipeline Interface

The core element of this bachelor thesis is the adjustable NLP pipeline that makes it possible to use (or not to use) certain NLP operations in a certain order. Using this technique in combination with an easy to use interface, it enables users to adjust the text processing to her or his needs. At the end of that processing, the user will have a compact summarizing representation and visual enhancements of the documents. Exploring and gaining insight into a large amount of different documents can happen significantly more quickly this way.

The NLP pipeline processes an array of text documents through individual NLP operations with selected parameters and passes it from one operation to the next one. Essentially, the entire pipeline serves as a means of preprocessing data for the resulting visualization. In addition, some steps of the pipeline depend on other steps. This sets up constraints for the order of pipeline operations. Operations like PoS tagging (and subsequent noun/verb/adjective selection) require tokenization, topic modeling within a single document requires segmentation. Figure 3.1 gives an overview of available NLP operations to be used in a pipeline as well as possible user input.

All pipeline operations are executed on each (input) text document. This approach generates "local" results per document. Such a document is usually the text content of a file or an entry of an RSS feed. Multiple files or an RSS feed containing multiple entries can be loaded to handle multiple documents, however. Each document can then be segmented into different parts, e.g. sentences, paragraphs or as-is (i.e. entire document as a whole). In case of RSS, the text content of each feed entry contains XML markup which needs to be removed from the content to avoid distortion (details in section 3.2).

There are dozens of possible configurations of the pipeline. Order and parameters of the individual pipeline operations can be changed or not set at all. The results are stored

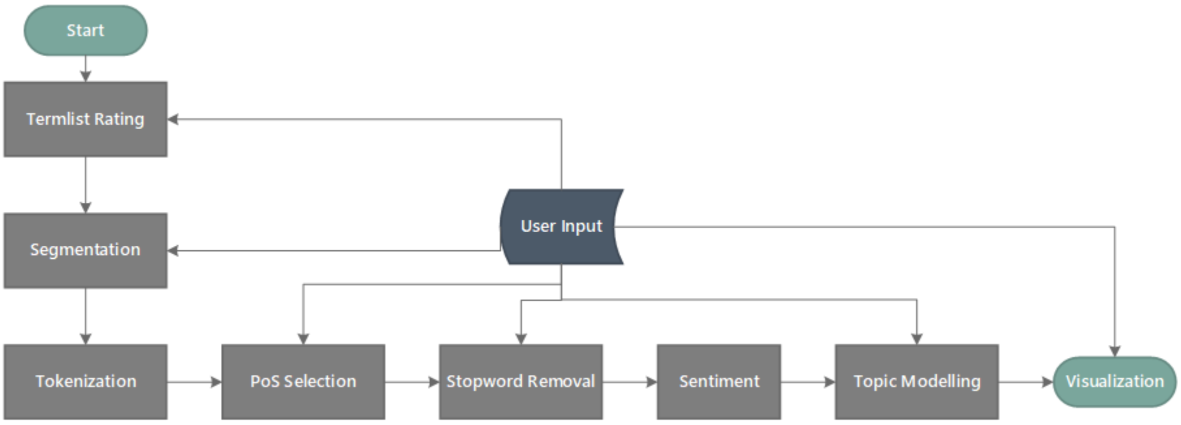


Figure 3.1: NLP Pipeline illustration. User input is available through parameters.

in-memory (in the order, which they have been executed) and displayed separately per input document. Optionally, a global summary takes place, which combines the previous local results of each document and performs topic modeling on them.

In the following sections, the individual pipeline steps will be described in more detail.

3.1 Termlist Rating

It is possible to manually enter a list of positive and a list of negative n-grams so a score can be calculated for how many positive and negative hits are in an document. This can be used to gather a quick overview of the importance of a text document and documents can be ordered according to that score. This step makes it possible to present the most important results first. It is independent from segmentation, tokenization and other following steps. It performs searches on the raw text content of a document so that steps like stopword removal and stemming cannot interfere with the results. Therefore entries in the list of terms can be unigrams or n-grams (i.e. multiple words). They can be set as parameters to termlist rating where two lists are available, one negative and one positive. List entries are separated by line break. Every entry is trimmed, removing redundant spaces at the end and beginning of every line.

Input Raw text documents

Parameters List of positive terms, list of negative terms

Output Score per input document

Constraints First pipeline element, if present

3.2 Segmentation

This is usually a very early step of the pipeline which takes place after termlist rating (if selected; otherwise it is the first step). This step is necessary for operations that require multiple documents to work with, like topic modeling. At first the given text will be separated into consecutive pieces based on user defined parameters and markup, if present, of the input. Segments can be sentences, paragraphs, or documents. The splitting level "paragraphs" is a step that requires markup (i.e. XML markup "<p>" and "</p>"), to be able to detect paragraphs. After this step is performed, the text content of every document is cleaned from that markup information. Sentence level splitting is performed on certain characters, such as ".", "!" and "?". During this step meta data and character sequences like line breaks are removed from the data. Empty segments are not included in the result. If segmentation is omitted as a whole then a default segmentation is performed at the first pipeline step that requires data in segmented form. It is equal to the option "document(s)" which leaves the input document(s) as-is. The output is a vector of documents, where each contains a vector of segments, where each segment contains text content without meta data. After segmentation, the document vector is referred to as corpus and denoted by $C = \{d_0, d_1, ..., d_n\}$ where $n \in \mathbb{N}$. Every document is then denoted by $d_x = \{s_0, s_1, ..., s_m\}$ where $m \in \mathbb{N}$.

Input Raw text documents (with markup)

Parameters Sentences, Paragraphs, or Documents

Output Documents vector containing segments vector without meta data

Constraints Paragraphs require markup

3.3 Tokenization

This process splits each of the segments further into tokens. Tokens are usually unigrams, which are essentially words in this work (but could also be letters, syllables, ...) [Gan14]. This is an implicit step and is performed at the first operation that requires it. After that step, all subsequent steps operate on term vectors. PoS selection and stopword removal require tokenization. To preserve information of results, segments are always organized in sentences, which are split the same way as described in Section 3.2. This also improves reusability since the same data structures can be used across the pipeline (see section 4.1.2 for details). Splitting is performed on certain characters, such as " ", "," and "-". If previous segmentation is set to "paragraphs" or "documents" then the sentence information is ignored. The extracted tokens are stored in-memory as term vectors to preserve their order, see Section 4.1.2 for details. This is contrary to the commonly used bag-of-words approach, in which order does not matter. From this point on, document segments are considered sets of tokens denoted by $s_x = \{t_0, t_1, ..., t_m\}$ where $m \in \mathbb{N}$ and $x \in \mathbb{N}$.

Input Segmented document vector

Parameters None

Output Documents vector containing segments vector containing a term vector

Constraints After segmentation

3.4 Part-of-Speech Selection

PoS selection allows the user to select nouns, verbs or adjectives from the text documents (i.e., filter all tokens that are not nouns, verbs or adjectives). The user can choose to select nouns, verbs or adjectives by setting parameters. These selections can be combined. So, if noun selection, verb selection and adjective selection are set as parameters, they are combined to select nouns, verbs and adjectives from the text. Part-of-Speech (PoS) tagging is required to determine the type of word so that it can be selected for further steps if it matches a filter and discarded otherwise. In addition to that, words can be highlighted based on their tag (see Section 3.8). PoS tagging is therefore implicitly applied at the first feature that requires it, such as this one. In PoS tagging, each token of the text segments is being checked against a base lexicon and a set of rules to determine its tag. The results of noun selection are visualized as described in Subsection 3.8.

Input Tokenized and segmented document vector

Parameters "Select" or "Highlight only" nouns, verbs or adjectives

Output Tokenized and segmented document vector containing tokens that are nouns

Constraints After tokenization

3.5 Stopword removal

Stopword removal filters a list of defined stopwords (for the English language) from the text stream. In addition, users can add a custom list of stopwords that will be removed as well.

Input Tokenized and segmented document vector or raw text stream

Parameters Custom list of stopwords

Output Tokenized and segmented document vector without stopwords

Constraints After tokenization

3.6 Sentiment Analysis

Sentiment analysis is performed per sentence based on the original term vector. The result, which consists of a sentiment score and label, is then stored (in-memory) per sentence as well. It is used for a brief summary per document to display a count of positive, neutral and negative sentiment labels. In addition, it is used for text highlighting at the resulting TextVis (see Section 3.8).

Input Tokenized and segmented document vector

Parameters None

Output Tokenized and segmented document vector with sentiment label and score per sentence

Constraints After tokenization

3.7 Topic Modeling

A core element to gain insight into what the text documents are about is topic modeling/extraction. Topic modeling techniques usually consider documents to be a combination of a limited number of topics. Topic extraction performs clustering to group several documents and tokens with similar properties. The results can be used to find sets of keywords that approximately represent the contents of a given document. To perform proper topic modeling, it is necessary to provide at least two documents or document segments as an input, otherwise there will not be meaningful results. Hence, for one document to be properly analysed, segmentation needs to take place to split the document into sentences or paragraphs (depending on user input). Only the contents of the local result (i.e. all pipeline operation results before topic modeling) of each document are taken into account for this step, due to implementation constraints as described in sections 4.2.4 and 4.2.3. Features like termlist rating are not considered for this task. Topic modeling is the last possible step to be performed in the NLP pipeline. It results either in a visualization or a text summary, depending on the selected approach due to implementation constraints. Two different approaches have been chosen for this project to perform topic modeling and extraction, as described below.

Input Vector of text strings that is based on (processed) segments

Parameters Number of topics, number of terms per topic, approach ("LDA" or "NMF")

Constraints After segmentation and tokenization

3.7.1 LDA

Latent Dirichlet Allocation (LDA) is a statistical topic model [MDH08]. In LDA based topic extraction mechanisms, every token or set of words can be associated with one of the specified topics. These topics can be labeled by terms, which will be selected based on the probability that a term is a assigned to a topic [Ost15]. In this project, the result of LDA is a list of maps which contains terms and probabilities to approximate the topic. The results of LDA are rendered in a simple list of these terms and probabilities next to the corresponding documents.

Output Keywords and scores per topic

3.7.2 NMF

Nonnegative matrix factorization (NMF) is a multi-variate analysis method used for clustering and classification of data [LYC10]. Term-document-matrices (TDMs) are high-dimensional non-negative feature matrices, which are perfectly suitable for this operation. Within this project NMF is applied for topic extraction and to determine the strength of links between estimated topics and documents (represented by the title of a document). The NMF topic modeling used in this project accepts an $m \times n$ TDM matrix. A TDM uses documents as column vectors and term occurrences as row vectors. A TDM can look like the following.

$$A = \begin{array}{c} \\ t_0 \\ t_1 \\ ... \\ t_m \end{array} \begin{array}{c} \begin{array}{cccc} d_0 & d_1 & ... & d_n \end{array} \\ \begin{pmatrix} 0 & 1 & ... & 1 \\ 1 & 1 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & 0 \end{pmatrix} \end{array}$$

In this example, the TDM uses only a binary metric to specify whether a document contains a term (1) or not (0). Other possible metrics are term frequency, which is a count of term occurences per document, and then there is tf-idf, which refers to "term frequency - inverse document frequency". It is used to give terms that appear in fewer documents a higher importance [KI15]. Depending on the prior segmentation, the tf-idf score is calculated either on terms in sentences, paragraphs or entire reports. This work uses tf-idf as weighting scheme for the DTM.

NMF factorizes the given $m \times n$ A matrix into two matrices W and H [LS00]. W is an $m \times r$-matrix and H is an $r \times n$-matrix. Matrix A can be approximated by linear combination of the columns of W with weights of the components of H, $A \approx WH$. Usually $r < m \vee r < n$, which leads to W and H matrices being smaller. That factorization is performed in multiplicative updates, which are executed until convergence or a fixed number of iterations is reached. The results of NMF contain a topic estimation, which is a map of keywords and scores per topic. In addition to that, an H matrix is created that determines the link strength between topics and the original documents.

A benefit of providing NMF in addition to LDA is increased variety and different results. According to Choo et al, NMF has an advantage over LDA when it comes to user feedback [CLRP13]. The local results of NMF are visualized in a horizontal bar chart that shows keywords within the bar and the width of the bar corresponds to the score of the topic.

Output Keywords and scores per topic, H matrix (link strength)

3.8 Text Rendering

To give a detailed visualization of a single processed document a text visualization takes place. Based on previously created results, selected sentences or tokens are given a markup to describe their visual appearance, see section 4.1.4 for further details.

Explicit PoS tagging operations always result in a visualization. Nouns have a green font color. Verbs use an orange font color. Adjectives are given a purple font color. Entire sentences can be highlighted as well. Sentiments for sentences are highlighted using a light green background color for a positive sentiment and light red for a negative sentiment. Terms that are contained in the positive termlist are highlighted with a cyan background. Terms, which are in the negative termlist are highlighted with a pink background. Sentences that are most representative of a topic, are highlighted with a light yellow background. These sentences are selected, using the sentence that is associated to the column with the maximum membership value (component) of the H matrix. Colors are chosen to have varying degrees of intensity/saturation to minimize overlap. A combined visualization of the aforementioned features is shown in Figure 3.2.

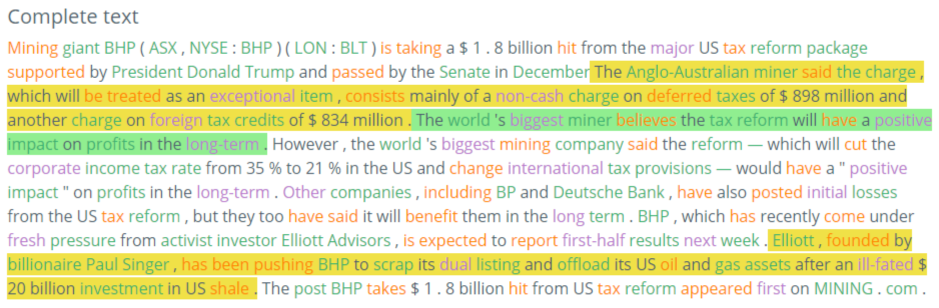
If multiple highlighting options are enabled in combination, then there is a priority by which terms are highlighted. First, sentences will be highlighted, based on topic. If there is no topic keyword contained within the sentence, then it will be highlighted based on sentiment. If there is no sentiment data on this sentence, it will not be highlighted at all (but individual tokens within the sentence still can). After sentence highlighting is complete, each token of the sentence will be checked and highlighted, if eligible. Tokens are highlighted, as previously described, based on their PoS tags and also whether they are part of a termlist hit. Tokens can have a font color according to their tag and a background color, which overrides the sentence's background color, if present, according to the termlist.

Input Tokenized and segmented document vector, topic keywords and scores

Parameters None

Output Text with markup for rendering in HTML

Constraints None



Complete text

Mining giant BHP (ASX , NYSE : BHP) (LON : BLT) is taking a $ 1 . 8 billion hit from the major US tax reform package supported by President Donald Trump and passed by the Senate in December The Anglo-Australian miner said the charge , which will be treated as an exceptional item , consists mainly of a non-cash charge on deferred taxes of $ 898 million and another charge on foreign tax credits of $ 834 million . The world 's biggest miner believes the tax reform will have a positive impact on profits in the long-term . However , the world 's biggest mining company said the reform — which will cut the corporate income tax rate from 35 % to 21 % in the US and change international tax provisions — would have a " positive impact " on profits in the long-term . Other companies , including BP and Deutsche Bank , have also posted initial losses from the US tax reform , but they too have said it will benefit them in the long term . BHP , which has recently come under fresh pressure from activist investor Elliott Advisors , is expected to report first-half results next week . Elliott , founded by billionaire Paul Singer , has been pushing BHP to scrap its dual listing and offload its US oil and gas assets after an ill-fated $ 20 billion investment in US shale . The post BHP takes $ 1 . 8 billion hit from US tax reform appeared first on MINING . com .

Figure 3.2: A fully rendered text with background color on sentences and font color on individual tokens.

3.9 Global Summary

If topic modeling using NMF or LDA has been performed, a global summary is created after the pipeline execution has finished. In case of NMF, the global summary will contain a visualization that represents the titles of the input documents and extracted keywords based on their content. To prepare the data for the visualization, a map of entities (document titles and topic keywords) has to be generated. The NMF output, as described in Section 3.7.2, containing the H matrix, is used to specify edges and their thickness. This kind of visualization presents the user an up front overview of all documents (represented by their titles) and all topics, as well as their association with them (see Figure 3.3 for an illustration and Section 4.1.5 for implementation related details). In case of LDA, the global summary will display a textual representation of global keyword and score pairs in a list (see Section 4.2.3 for details). The global LDA based result is shown in Figure 3.4.

Input Tokenized and segmented document vector for all documents, global topic keywords and scores

Parameters None

Output Visualization

Constraints None

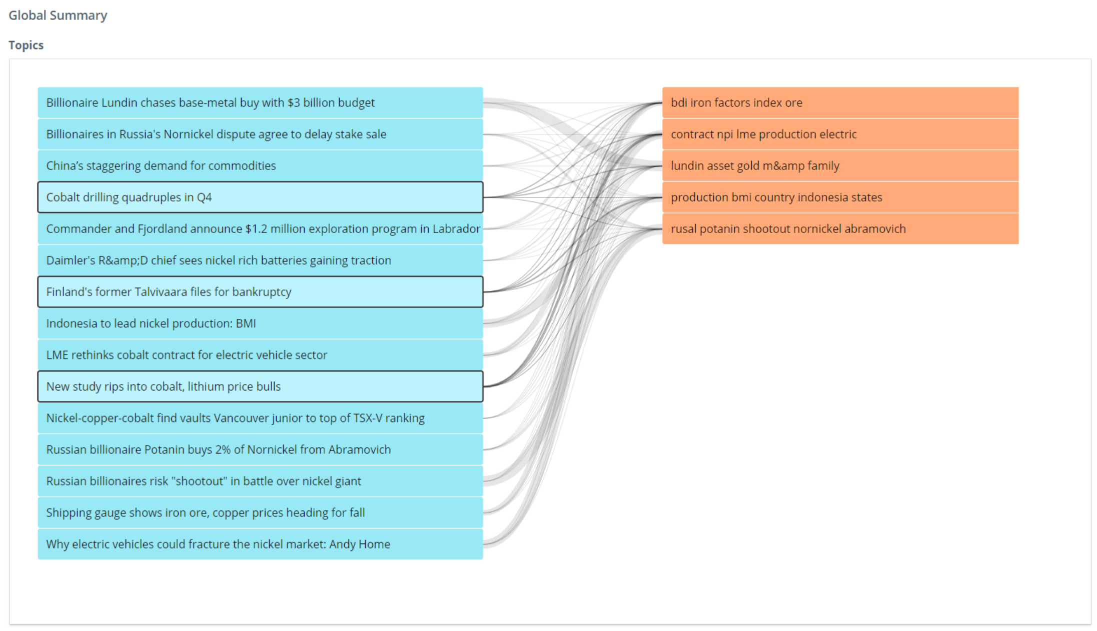
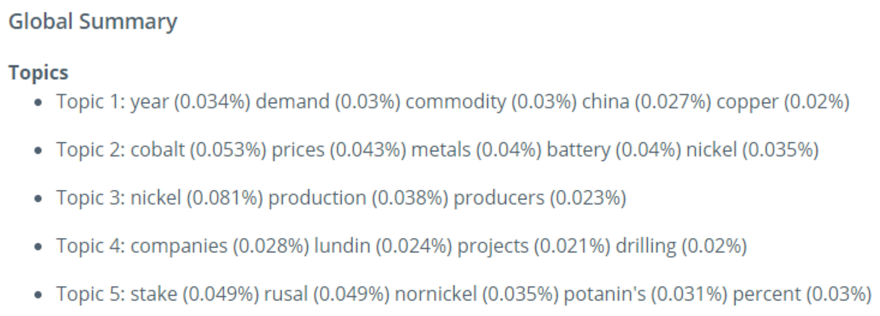

Global Summary
Topics
Billionaire Lundin chases base-metal buy with $3 billion budget
Billionaires in Russia's Nornickel dispute agree to delay stake sale
China's staggering demand for commodities
Cobalt drilling quadruples in Q4
Commander and Fjordland announce $1.2 million exploration program in Labrador
Daimler's R&amp;D chief sees nickel rich batteries gaining traction
Finland's former Talvivaara files for bankruptcy
Indonesia to lead nickel production: BMI
LME rethinks cobalt contract for electric vehicle sector
New study rips into cobalt, lithium price bulls
Nickel-copper-cobalt find vaults Vancouver junior to top of TSX-V ranking
Russian billionaire Potanin buys 2% of Nornickel from Abramovich
Russian billionaires risk "shootout" in battle over nickel giant
Shipping gauge shows iron ore, copper prices heading for fall
Why electric vehicles could fracture the nickel market: Andy Home
bdi iron factors index ore
contract npi lme production electric
lundin asset gold m&amp family
production bmi country indonesia states
rusal potanin shootout nornickel abramovich

Figure 3.3: A global summary visualization using NMF. 15 document titles are on the left. 5 topics with 5 keywords each are on the right. Edges represent link strength between documents and topic keywords.


Global Summary

Topics

- Topic 1: year (0.034%) demand (0.03%) commodity (0.03%) china (0.027%) copper (0.02%)
- Topic 2: cobalt (0.053%) prices (0.043%) metals (0.04%) battery (0.04%) nickel (0.035%)
- Topic 3: nickel (0.081%) production (0.038%) producers (0.023%)
- Topic 4: companies (0.028%) lundin (0.024%) projects (0.021%) drilling (0.02%)
- Topic 5: stake (0.049%) rusal (0.049%) nornickel (0.035%) potanin's (0.031%) percent (0.03%)

Figure 3.4: A global summary visualization using LDA, represented in an unordered list containing 5 topics with 5 keywords each and a score for each keyword within parentheses.

CHAPTER 4

Implementation

The implementation has been carried out using web technologies, primarily TypeScript and JavaScript. Due to requirements of the stakeholders of this project, the entire computation has to be carried out within the local client and must not transfer any document content to or via an external network.

4.1 Client

The web client is designed to run in most common web browsers (Chrome, Firefox, Edge). It itself requires a web server to be hosted and be accessible. Within the client a UI that has been build from ground-up using state of the art components namely Angular (including RxJS), Bootstrap, jQuery and d3 (see Section 4.2 for further details).

The first visible artifact that appears in the browser is a login screen, as can be seen in Figure 4.1. After a successful login the main UI appears (see Figure 4.2). The main UI is designed like a dashboard. It provides multiple cards that contain information and are arranged in a grid. In this view the card component "report loader" is the first card that is present. It can be used to load documents from a data source. Further cards are added and removed dynamically, when the user loads reports from a data source. If the user wants to load different reports from a different data source than the currently loaded documents, the currently loaded document cards will be removed and new ones, for the new documents, will be created. The cards can be rearranged dynamically as well, if order information is presented, which can be created using termlist rating.

After documents have been loaded the card "NLP pipeline" appears and a card for each individual report/document as shown in Figure 4.3. To see the full text content of a document, the "Expand" button has to be clicked on the card of a document. A dialog containing the full text content is then displayed, as can be seen in Figure 4.4.

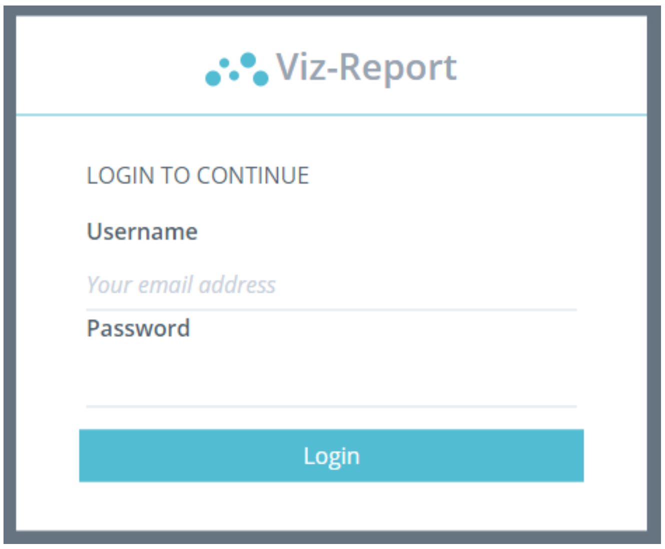


Figure 4.1: Login Page.

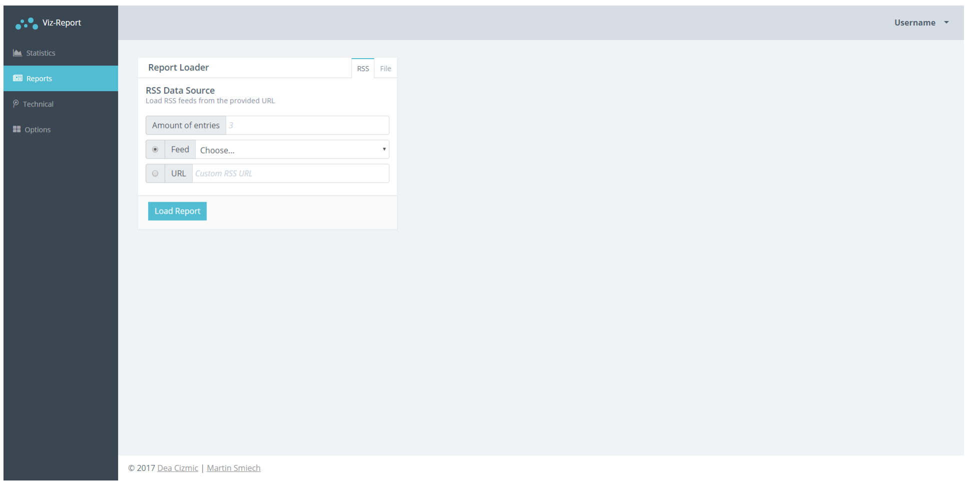
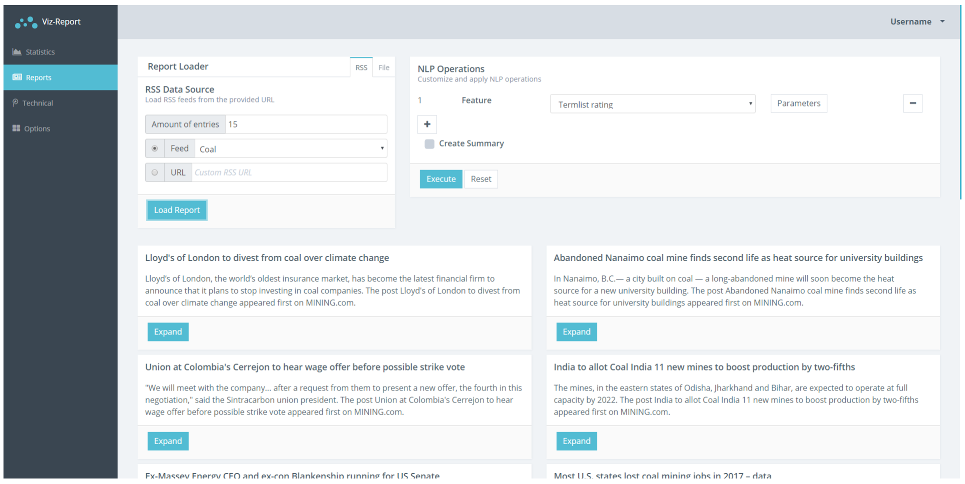


Figure 4.2: Report Page without reports.



Lloyd's of London to divest from coal over climate change

Lloyd's of London, the world's oldest insurance market, has become the latest financial firm to announce that it plans to stop investing in coal companies. The post Lloyd's of London to divest from coal over climate change appeared first on MINING.com.

Abandoned Nanaimo coal mine finds second life as heat source for university buildings

In Nanaimo, B.C.— a city built on coal — a long-abandoned mine will soon become the heat source for a new university building. The post Abandoned Nanaimo coal mine finds second life as heat source for university buildings appeared first on MINING.com.

Union at Colombia's Cerrejon to hear wage offer before possible strike vote

"We will meet with the company... after a request from them to present a new offer, the fourth in this negotiation," said the Sintracarbon union president. The post Union at Colombia's Cerrejon to hear wage offer before possible strike vote appeared first on MINING.com.

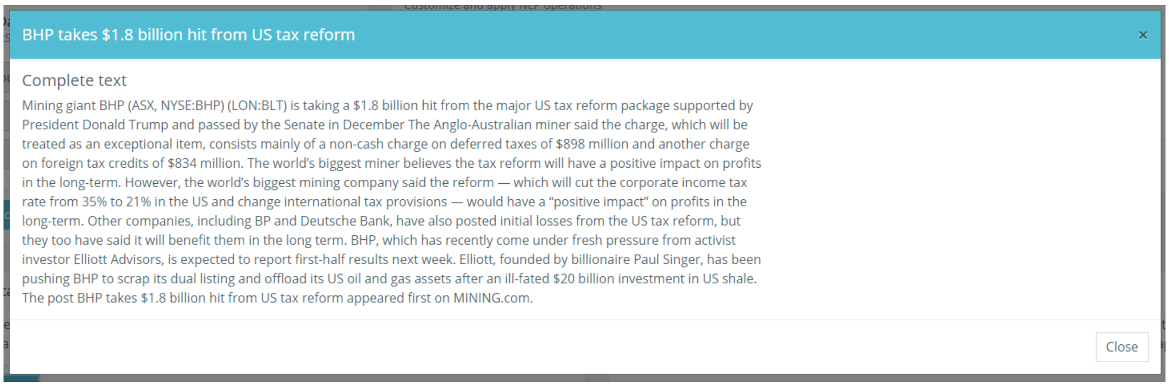
India to allot Coal India 11 new mines to boost production by two-fifths

The mines, in the eastern states of Odisha, Jharkhand and Bihar, are expected to operate at full capacity by 2022. The post India to allot Coal India 11 new mines to boost production by two-fifths appeared first on MINING.com.

Ex-Massey Energy CEO and ex-con Blankenship running for US Senate

Most U.S. states lost coal mining jobs in 2017 – data

Figure 4.3: Report Loader.



BHP takes $1.8 billion hit from US tax reform

Complete text

Mining giant BHP (ASX, NYSE:BHP) (LON:BLT) is taking a $1.8 billion hit from the major US tax reform package supported by President Donald Trump and passed by the Senate in December The Anglo-Australian miner said the charge, which will be treated as an exceptional item, consists mainly of a non-cash charge on deferred taxes of $898 million and another charge on foreign tax credits of $834 million. The world's biggest miner believes the tax reform will have a positive impact on profits in the long-term. However, the world's biggest mining company said the reform — which will cut the corporate income tax rate from 35% to 21% in the US and change international tax provisions — would have a "positive impact" on profits in the long-term. Other companies, including BP and Deutsche Bank, have also posted initial losses from the US tax reform, but they too have said it will benefit them in the long term. BHP, which has recently come under fresh pressure from activist investor Elliott Advisors, is expected to report first-half results next week. Elliott, founded by billionaire Paul Singer, has been pushing BHP to scrap its dual listing and offload its US oil and gas assets after an ill-fated $20 billion investment in US shale. The post BHP takes $1.8 billion hit from US tax reform appeared first on MINING.com.

Figure 4.4: Detailed view of document content without pipeline results.

At this point, the user can create a custom NLP pipeline in a custom order with custom elements. A fully configured pipeline card usually looks like what is shown in Figure 4.5. When this pipeline is executed, there are three kinds of results. One are brief results, which are displayed within the document cards below the brief text summary of every document. Then there is a detailed result, which is visible in a dialog that appears when clicked on "Expand" of a card. The detailed result is shown as in Figure 4.7. On the left of this dialog, there are interactive controls to manipulate the visualization in real time. The last way to show results is the global result, which provides a summary across all loaded documents, if selected and processed accordingly. The global result can be seen in Figures 3.3 for NMF and 3.4 for LDA.

4.1.1 Data Sources

The application supports two data sources from which documents can be loaded. The first one are RSS Feeds and the second one are plain text files. RSS feeds contain metadata about author, date as well as structural information like paragraphs, which are used for segmentation. That metadata is in XML. XML markup needs to be removed for further processing and display. To retrieve these RSS feeds, rss2json.com web service is used. This service enables easy conversion from RSS feeds to JSON strings, which can be conveniently used in the web client implementation. The drawback of rss2json is that it limits the number of results to 15 in its free version (at the time of writing). As requested by the stakeholders, the application contains a list of predefined RSS URLs for all available types of commodities at mining.com.

In contrast, text files that are supported have none of these metadata tags. To support date and title information within these plain text documents, a check is performed on the first line whether it contains a "|" character. If this is the case, then the part of the line before the "|" character is checked for a date, if so, it is used as date of the document. The second half of the line will be used as title. This is a custom parsing method which is adjusted to data provided by stakeholders. The supplied data was automatically crawled from non-RSS mining.com articles, which are a preferred data source from this work's stakeholders.

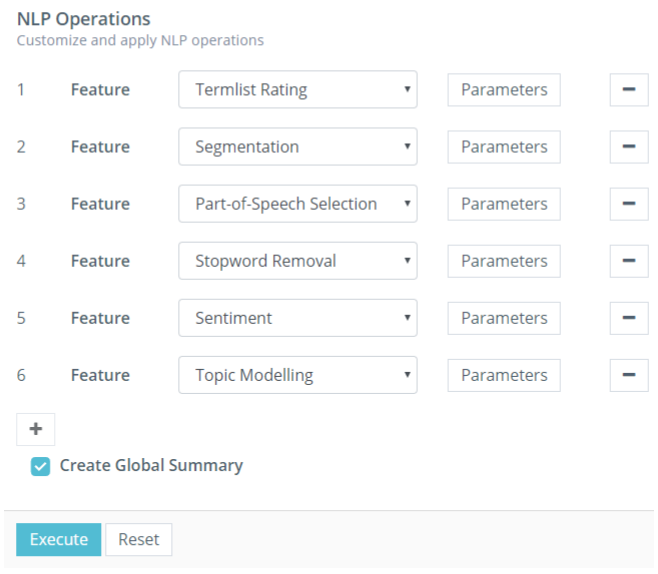
4.1.2 Data Structures

During segmentation and tokenization, the data is transformed from a raw text string into multiple nested arrays. At first, there is an array which is organized by documents. Before segmentation and tokenization, every element in this array contains a string. After segmentation, every element of that array contains another array, which is organized by segments. If segmentation is set to "Documents", then a document array element contains exactly one (segment) array. Every segment array contains an array of sentences that are included in it. A sentence is a JavaScript object, which is based on compendium-js's result structure (see Section 4.2.2 for further details). Every sentence object has a property "tokens", which is an array of token objects.

Since some operations like topic modeling require a string or string array as input, a string content array of the segments is maintained in parallel (per document). This string array is updated based on filter operations on the aforementioned data structure, which is derived from compendium-js's results. This is a means of caching results to improve reusability and performance within the application. This is because results can be required more than once, e.g. once for a local summary and once for a global summary.

4.1.3 Pipeline Configuration Card

In the actual implementation, the features "Noun Selection", "Verb Selection" and "Adjective Selection" are binary parameters for PoS Selection. These parameters can be set via checkbox in the parameter dialog for PoS Selection. The parameter dialog can be seen in Figure 4.6. PoS Selection requires PoS tagging to take place. PoS tagging is necessary to determine the tag for a term. There are two flags for each of these features, namely "nouns", "verbs" and "adjectives", as well as whether it is "selection" or "highlighting only. See Figure 4.5 for reference. This is due to compendium-js behaviour, which has been used to perform the three aforementioned selection operations, see Section 4.2, Subsection 4.2.2 for further details. The highlighting option can be changed at the details view to enable and disable visualization with instant results, as shown in Figure 4.7.

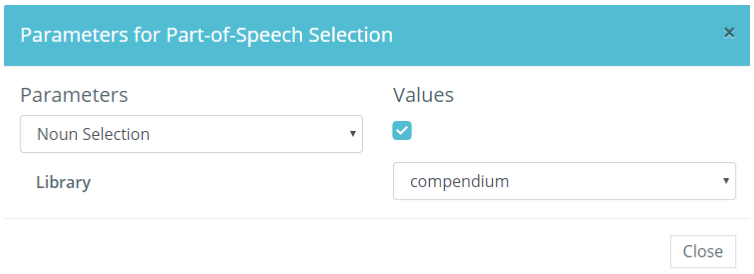

NLP Operations
Customize and apply NLP operations

1	Feature	Termlist Rating	Parameters	−
2	Feature	Segmentation	Parameters	−
3	Feature	Part-of-Speech Selection	Parameters	−
4	Feature	Stopword Removal	Parameters	−
5	Feature	Sentiment	Parameters	−
6	Feature	Topic Modelling	Parameters	−

+

✓ Create Global Summary

Figure 4.5: Fully configured NLP pipeline card.

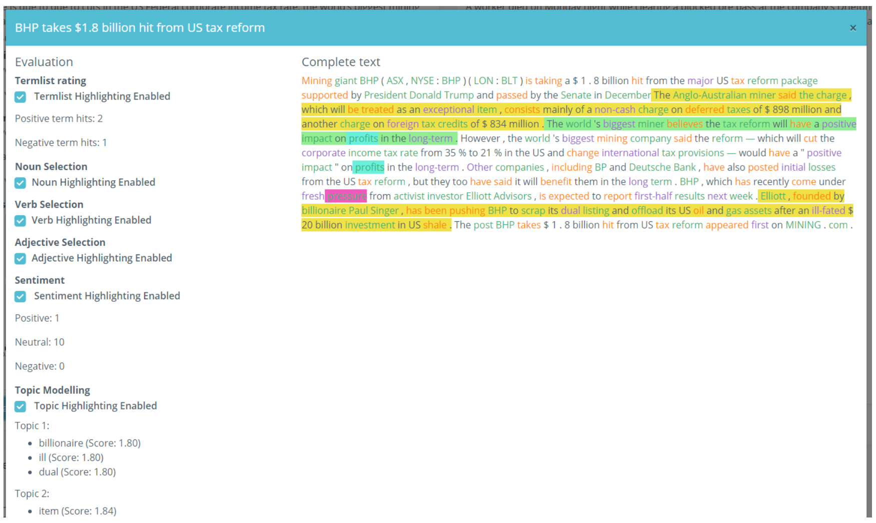

Parameters for Part-of-Speech Selection

Parameters	Values
Noun Selection	✓
Library	compendium

Figure 4.6: Parameter dialog for PoS Selection.

4.1.4 Text Rendering Implementation

To perform text rendering, the native HTML element "span" is used with a class attribute to describe the visual appearance of one or more tokens. As described in Section 3.8, the application iterates over every segment, and every sentence, and every token within it. Segments are not relevant for the text rendering implementation. If sentences have sentiment data or contain a keyword of a topic, then a span-tag with the corresponding class attribute is wrapped around the sentence. This class is described in CSS, to either have a yellow background color for topic keywords or a red or green background for negative or positive sentiment, respectively. A combination of all text rendering options can be seen in Figure 4.7.


BHP takes $1.8 billion hit from US tax reform

Evaluation

Termlist rating

✓ Termlist Highlighting Enabled

Positive term hits: 2

Negative term hits: 1

Noun Selection

✓ Noun Highlighting Enabled

Verb Selection

✓ Verb Highlighting Enabled

Adjective Selection

✓ Adjective Highlighting Enabled

Sentiment

✓ Sentiment Highlighting Enabled

Positive: 1

Neutral: 10

Negative: 0

Topic Modelling

✓ Topic Highlighting Enabled

Topic 1:

- billionaire (Score: 1.80)
- ill (Score: 1.80)
- dual (Score: 1.80)

Topic 2:

- item (Score: 1.84)

Complete text

Mining giant BHP (ASX , NYSE : BHP) (LON : BLT) is taking a $ 1 . 8 billion hit from the major US tax reform package supported by President Donald Trump and passed by the Senate in December The Anglo-Australian miner said the charge , which will be treated as an exceptional item , consists mainly of a non-cash charge on deferred taxes of $ 898 million and another charge on foreign tax credits of $ 834 million . The world 's biggest miner believes the tax reform will have a positive impact on profits in the long-term . However , the world 's biggest mining company said the reform — which will cut the corporate income tax rate from 35 % to 21 % in the US and change international tax provisions — would have a " positive impact " on profits in the long-term . Other companies , including BP and Deutsche Bank , have also posted initial losses from the US tax reform , but they too have said it will benefit them in the long term . BHP , which has recently come under fresh pressure from activist investor Elliott Advisors , is expected to report first-half results next week . Elliott , founded by billionaire Paul Singer , has been pushing BHP to scrap its dual listing and offload its US oil and gas assets after an ill-fated $ 20 billion investment in US shale . The post BHP takes $ 1 . 8 billion hit from US tax reform appeared first on MINING . com .

Figure 4.7: Fully rendered text with all highlighting options enabled.

4.1.5 Global NMF Visualization

For the global visualization of NMF results, a parallel list with edges has been implemented using d3. It is based on the BiSet Edge Bundeling [SMNR16] implementation by Lukas Eibensteiner and Manuel Kapferer [Eib17] and adaptations made for topic modeling by Michael Mazurek [MW18]. These adaptations remove (bi)clustering and edge bundling, since they are not needed to represent the documents, topics and the relationship between them.

4.1.6 Default Settings and Values

The whole user interface as well as each pipeline step has default settings or values to make usage easier and reduce initial configuration time. These settings and values are listed here. For document loading a default setting is set to the predefined mining.com RSS feeds, as described in Section 4.1.1. This list of RSS feeds is available to the users via a dropdown menu. The default number of entries to load from an RSS feed is 3. Regardless of rss2json, mining.com only provides up to 15 entries. The default splitting level for segmentation is set to "Sentences". Topic modeling has 2 as topic count and 3 as terms each by default. Every highlighting option for text rendering/visualization is set to "true" by default.

4.2 Frameworks and Libraries

To achieve a state-of-the-art implementation of the aforementioned methodology, a number of libraries and frameworks is used.

4.2.1 UI Frameworks and Libraries

To provide an interactive and reliable user interface that is compatible with various web browsers, the Angular (ng) web framework (v4) has been chosen. The angular framework embraces a multi-paradigm programming approach that is both functional and reactive (FRP) on top of being object-oriented [Jac16]. This framework imposes an architecture design on this project that is regarded state-of-the-art. Furthermore, this brings a lot of advantages to the development of the herein described project. The angular framework brings a whole collection of libraries and features like Angular Components, RxJS and Observables. Angular Components provide an efficient way of managing view templates and storing related logic/code in separate containers. Using RxJS and Observables, changes in data are propagated automatically, so when a variable changes all other occurrences (like functions) of the variable get updated/re-executed as well.

The chosen angular framework version uses TypeScript (v2) as its language, which is a substantial enrichment for developers when compared to regular JavaScript (ES5) that's commonly used in browsers. The selected version of TS allows static type safety and a higher degree of abstraction for web applications when compared to common JavaScript (ES5) [GBB17]. In this project, the benefits of using TypeScript outweigh the drawbacks. One major benefit is improved robustness to the application. The adjustable NLP pipeline is heavily focused around user interaction and user input. Using TypeScript in combination with Angular components it makes it easier and more readable to create a proper UI for manipulating and setting steps of the pipeline [GBB17]. It also improves recycling of code through abstraction (e.g., using interfaces). Another major benefit is improved error detection and handling which allows to find common UI bugs more quickly or prevent them entirely by design.

The project uses Twitter Bootstrap 4, the most recent version of Bootstrap. Bootstrap offers a streamlined collection of UI elements and styles for the browser that follow general design principles and strive look appealing to users [Ott17]. It features responsive design to enable size and resolution independent display of the website on multiple different devices. Mobile support is also present. Grid based alignment systems and standardized size properties aid creating an easy to follow and coherent user interface. In combination with Angular, view templates fit into the standard look and feel of Bootstrap [BWSP13]. Bootstrap features like (modal) dialogues and menus are used to display data stored in models inside Angular seamlessly and updated in real-time.

As a template for Bootstrap 4, the theme ModularAdmin has been chosen and applied [VH17]. ModularAdmin is an open source (and MIT licensed) theme for Bootstrap 4 and contains a card-oriented dashboard design. Unfortunately, it is not designed for Angular and had to be adapted. ModularAdmin's templates were written using Handlebars.js (HBS) template syntax, which had to be translated to angular template syntax and wrapped in angular components. Angular also imposes stricter rules regarding access to variables and other components (e.g. from inside the template code), which had to be respected during the transition to angular. Unnecessary features like the included JavaScript chart libraries have been removed, since d3 is used in this work. This template has been chosen because it offers a standard layout and design that can be reused and built upon. In combination with angular, it aids at a structured and standardized development process and reduces engineering workload throughout the project (i.e. in long term), especially when compared to writing everything from scratch. Another reason for this template is that it was accepted (and favoured) by this project's stakeholders.

4.2.2 compendium-js

Compendium-js is a natural language processing library written in JavaScript that performs PoS tagging, stemming and sentiment analysis on given input texts [Lau18]. In this project the version 0.0.28 of compendium-js is used. Compendium-js has been chosen because it delivers all results up-front in a data structure that is (re)used for most of the pipeline steps. The result's data structure is organized around sentences and tokens. It is an array of sentences, where each sentence has a set of properties including sentiment and tokens. The (JavaScript) object property "tokens" contains an array of tokens. Every token object contains the following data: raw term, stem, PoS tag and various attributes like whether it is an acronym, plural. Compendium-js tokenizes on term level. For additional structural information, the compendium-js results are performed per segment (depending on segmentation as described in Section 3.2) and then stored in an array which resembles the segments.

Compendium-js uses a lexicon which consists of 10,000 most common English words and a list of sentiment scores [Lau18]. Its PoS tagger achieves a score of 92% when tested against the Penn Treebank reference corpus. It is based on a Brill tagger, which is a simple, yet very effective rule based PoS tagger [Bri92]. In contrast to a statistical tagger, which picks tags based on probabilities, this tagger determines the tag of a term using a set of logical rules. These rules can be thought of like if-then statements, e.g. "If a term has a tag x in context C, then set that tag to y, or ...". The tags of a term are usually adjusted throughout the tagging process.

4.2.3 lda

LDA is a JavaScript implementation of latent dirichlet analysis (LDA) to extract topic information from a given document [Bec17]. This library has a simple API that has one function called "lda" with 3 parameters: document, topic count and number of terms per topic. The first parameter, document, is a simple text string. Therefore, if previous tokenization took place, the term vector is joined back together into a text string (respecting segmentation). The result is an array containing the topics and within the topics there are terms and corresponding scores. At first this LDA library was the primary element of topic modeling but this focus has been shifted towards nmf-js since the LDA library lacks flexibility and does not offer information about the link strength between keywords and the original documents. Due to this lack of relationship data, the options for visualization are limited and the result is presented as a list with topic keywords and scores. LDA is still included as a library for topic modeling in the implementation.

4.2.4 nmf-js

The JavaScript library nmf-js performs non-negative matrix factorization (NMF) [Ane13]. It provides a function that performs multiplicative updates called "mu" which accepts a $n \times m$-Matrix (DTM in this case), number of topics and terms to be extracted, a tolerance value and a maximum number of iterations as parameters. The tolerance value is set to 0.001 and the maximum number of iterations is capped at 75. The higher the number of iterations, to longer topic modeling can take, if convergence is not reached before.

4.2.5 d3

Data-Driven-Documents (d3) is a commonly used web visualization library, which uses the standard web technologies HTML, SVG, CSS and JavaScript to visualize data [BOH11]. It is applied for the herein described project to visualize the results of NMF topic modeling. The visualization is used as described in Sections 3.7.2 and 3.9.

CHAPTER 5

Evaluation and Results

A set of tests and evaluations has been performed to establish a comparative basis for the outcome of this project. During this process, the software was has been tested against a defined specification (i.e., pipeline configurations and documents). Some of the artefacts produced by the software have been presented to different users and their responds has been documented. In addition to that, performance of the application was measured.

5.1 Test Configuration and Input

To illustrate variety and differences in configuration and results, three pipeline examples and their results for three document are provided (see appendix). These configurations, documents and results are used as a reference throughout the evaluation.

Pipeline 1:

1. Segmentation
    a) Splitting level: "sentence"
2. Topic Modeling
    a) Topics: 3
    b) Terms each: 4
    c) Library: NMF

Pipeline 2:

1. Segmentation

   a) Splitting level: "sentence"

2. Stopword Removal
3. Topic Modeling
   a) Topics: 3
   b) Terms each: 4
   c) Library: NMF

Pipeline 3:

1. Segmentation
   a) Splitting level: "sentence"
2. PoS Selection
   a) Noun Selection
   b) Adjective Selection
3. Topic Modeling
   a) Topics: 3
   b) Terms each: 4
   c) Library: NMF

5.2 User Feedback

To encompass feedback, two different surveys have been carried through with a different audience each. The first audience is a general audience (lay persons), which has been issued a structured feedback form. This feedback form contained a set of documents and a set of results for each document. The users had to read through three documents and look at the three different application outputs for each document (see Appendix 6) and rate them using a 5 point Likert scale (from 1 = "strongly agree" to 5 = "not agree at all"). The results have been created using the three previously defined pipelines. The test documents have been chosen to be long (between 1000 and 2000 words each) but as little domain specific as possible. Since the target audience is not directly reachable but only through a stakeholder, it is not possible to get exact feedback from traders. Because of that, the four surveyed users are from a general audience. The results were presented to users who then answered the prepared questions regarding the outcome. Users were asked how much they agree on the representational quality of the results in relation to the input texts. The collected results are presented in Tables 5.1, 5.2 and 5.3.

Document 1	Score Pipeline 1	Score Pipeline 2	Score Pipeline 3
Person 1	3	2	4
Person 2	3	2	4
Person 3	4	3	3
Person 4	3	3	1

Table 5.1: User scores for document 1.

Document 2	Score Pipeline 1	Score Pipeline 2	Score Pipeline 3
Person 1	1	3	3
Person 2	2	3	4
Person 3	2	4	4
Person 4	2	3	3

Table 5.2: User scores for document 2.

Document 3	Score Pipeline 1	Score Pipeline 2	Score Pipeline 3
Person 1	2	3	2
Person 2	2	3	4
Person 3	4	4	2
Person 4	1	2	3

Table 5.3: User scores for document 3.

Documents	Average Score Pipeline 1	Average Score Pipeline 2	Average Score Pipeline 3
Document 1	3.25	2.5	3.0
Document 2	1.75	3.25	3.5
Document 3	2.25	3.0	2.75

Table 5.4: Average user scores across all three documents.

To sum it up, the average scores for each document are shown in Table 5.4. A visual representation of the results can be seen in Figure 5.1.

The outcome of this survey suggests that pipeline 2 works slightly better than pipeline 1 and 3 for document 1. Document 2 appears to work best with pipeline 1 , which does neither PoS selection nor stopword removal. For document 3 the pipeline 1 also seems to work slightly better than 3, which seems to work slightly better than pipeline 2. Pipeline 1, which lacks preprocessing almost entirely (except segmentation and therefore markup removal), appears to perform best (in 2 out of 3 cases) according to surveyed audience. Examples like document 1, indicate, that preprocessing can improve the quality of results. Overall, there is no clear winner, however. In the cases of document 2 and 3 the combination of the preprocessing and the selected weighting scheme tf-idf seem to worsen the results. In such cases, a different weighting scheme, such as simply term

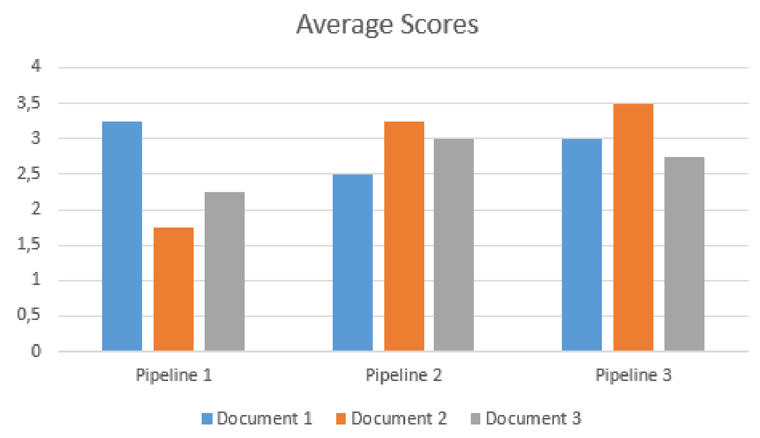


Figure 5.1: Bar chart representation of the survey results (the lower, the better).

frequency (tf), could improve results.

The survey also contained an open question at the end, which was "How do you think these keywords could be useful?". Among the answers to the question were suggestions to use it to create "links to other articles". However, a major issue on the user side (by 3 out of 4 participants) was that they cannot imagine that these keywords could be useful to them. Beyond that, one user noted that critical keywords were missing in her opinion, which were "rotten tomatoes" for document 1 and "ceasefire" for document 2. This lack can be explained due to the high frequency of these keywords, which are neglected because of the tf-idf weighting scheme and can be improved, as described before, by using tf as a weighting scheme.

In addition to the previously described lay person survey (general audience), an informal review with the stakeholders has been conducted to gather further feedback. During this review, the software has been presented to the stakeholders, who then described their experience with it and shared their opinion about the application. In general, the application is helpful and very appealing to companies in the commodity market. The review suggested that the results do not replace the need for reading the documents entirely, but aid at focusing on specific ones.

5.3 Performance

To provide an insight about performance a few executions have been measured for their duration. These tests have been carried out on a Windows 10 64-Bit machine (with all updates until 2018-02 installed, including Meltdown/Spectre fix) within Google Chrome version 63 (64-Bit). The hardware used was an Intel Core i7 7700HQ with 16GB DDR4 RAM on a 1920x1080 resolution display.

Note that these numbers have been measured using debug output from a development

Documents	Pipeline 1	Pipeline 2	Pipeline 3
Survey (all three documents)	5.5 seconds	5.7 seconds	7.5 seconds

Table 5.5: Measured performance for all used pipelines and documents.

environment. To provide one more example using more files, a pipeline execution consisting of "pipeline 2" on 20 documents of roughly the same size as the reference documents provided, takes 17,8 seconds.

All of these times are perceived as slow and are clearly noticeable to the user, which has been discussed in the aforementioned review as well. Purely web based approaches are limited technologically. On the one hand, web browsers have multiple layers of abstraction since the application has to be hardware independent. On the other hand, as mentioned in Section 2.1, most reference implementations are written in other languages or environments, such as Python. The latter case limits the variety of libraries to choose from and related works for references. These limitations could not be overcome in this work since this would violate the constraints of the stakeholders.

CHAPTER 6

Conclusion

Based on the previously presented outcomes, the pipeline configuration including segmentation (on sentence level), no PoS selection, stopword removal and NMF for topic modeling (i.e., pipeline 2 in Section 5.2) seems to work better than the pipeline without stopword removal and PoS selection instead. According to the feedback, users seemed to lean towards the pipeline without any preprocessing. Another finding revealed that PoS selection costs quite some performance and does not benefit the quality of results.

In the end, however, there is no universal method that suits all different documents and domains, hence the user can adjust and tweak pipeline configurations to find results that fit the given documents best. It aids at finding documents and document sections that are relevant more quickly and avoiding ones that are not relevant.

Using keywords to represent topics and what a document is about may not be enough to give the user a good understanding about certain documents. A more sufficient representation and summarization approach would be to present the sentences that are most representative for a topic/keyword.

To improve performance and gain a bigger variety of reference implementations, a substantial change in the software architecture would be necessary. For example, a Python server could be set up to perform NLP operations and the web front end could only be used to display results. For this, however, the user would have to send her or his data to a remote server or install and run a server locally.

For future work in the commodity domain, a more advanced topic modeling approach, that supports training of domain specific data is a desired feature. Individual preprocessing steps could be expanded for better support of n-grams, enabling, for example, custom weighting mechanisms for individual n-grams. Another possible expansion is to aggregate documents with specific contents by date to detect possible events that occurred in a specific topic at a specific time. Also, deep exploration and analysis for which NLP pipeline works best would be an important future work to deliver tangible comparisons.

List of Figures

List of Tables

Bibliography

[Ane13] Aneesha. nmf.js. https://github.com/aneesha/nmf.js, Dec 2013. Accessed: 2018-02-28.

[awe18] keon/awesome-nlp. https://github.com/keon/awesome-nlp, Jan 2018. Accessed: 2018-02-28.

[Bec17] Kory Becker. Lda topic modeling for node.js. https://github.com/primaryobjects/lda, Sep 2017. Accessed: 2018-02-28.

[BOH11] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, Dec 2011.

[Bri92] Eric Brill. A simple rule-based part of speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, ANLC '92, pages 152–155, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.

[BWSP13] V. Balasubramanee, C. Wimalasena, R. Singh, and M. Pierce. Twitter bootstrap and angularjs: Frontend frameworks to expedite science gateway development. In *2013 IEEE International Conference on Cluster Computing (CLUSTER)*, pages 1–1, Sept 2013.

[CLRP13] J. Choo, C. Lee, C. K. Reddy, and H. Park. Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1992–2001, Dec 2013.

[cor] Corenlp server. https://stanfordnlp.github.io/CoreNLP/corenlp-server.html. Accessed: 2018-02-28.

[Eib17] Lukas Eibensteiner. Biset implementation. https://www.cg.tuwien.ac.at/courses/Visualisierung2/HallOfFame/2017/Sun2015/html/index.html, May 2017. Accessed: 2018-02-28.

[Gan14] Kavita Ganesan. Text mining, analytics and more. http://text-analytics101.rxnlp.com/2014/11/what-are-n-grams.html, Nov 2014. Accessed: 2018-02-28.

[GBB17] Z. Gao, C. Bird, and E. T. Barr. To type or not to type: Quantifying detectable bugs in javascript. In *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pages 758–769, May 2017.

[Jac16] Luka Jacobowitz. Functional reactive programming in angular 2. `http://lukajcb.github.io/blog/angular2/2016/04/02/frp-in-angular-2.html`, Apr 2016. Accessed: 2018-02-28.

[JZ16] Xinyi Jiang and Jiawan Zhang. A text visualization method for cross-domain research topic mining. *Journal of Visualization*, 19(3):561–576, Aug 2016.

[KI15] B. A. Kuncoro and B. H. Iswanto. Tf-idf method in ranking keywords of instagram users' image captions. In *2015 International Conference on Information Technology Systems and Innovation (ICITSI)*, pages 1–5, Nov 2015.

[KK15] K. Kucher and A. Kerren. Text visualization techniques: Taxonomy, visual survey, and community insights. In *2015 IEEE Pacific Visualization Symposium (PacificVis)*, pages 117–121, April 2015.

[Lau18] Xav Laumonier. compendium-js. `https://github.com/Ulflander/compendium-js/wiki/analysis-process`, Jan 2018. Accessed: 2018-02-28.

[LRR11] M. Lobur, A. Romanyuk, and M. Romanyshyn. Using nltk for educational and scientific purposes. In *2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, pages 426–428, Feb 2011.

[LS00] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*, NIPS'00, pages 535–541, Cambridge, MA, USA, 2000. MIT Press.

[LYC10] H. Lee, J. Yoo, and S. Choi. Semi-supervised nonnegative matrix factorization. *IEEE Signal Processing Letters*, 17(1):4–7, Jan 2010.

[MDH08] Anuj Mahajan, Lipika Dey, and Sk. Mirajul Haque. Mining financial news for major events and their impacts on the market. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '08, pages 423–426, Washington, DC, USA, 2008. IEEE Computer Society.

[MDP$^+$11] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 142–150, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[MSB$^+$14] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[MW18] Michael Mazurek and Manuela Waldner. Visualizing query expansion results. *to appear: Computer Graphics Forum 2018*, 2018.

[Ope] Apache opennlp manual. `https://opennlp.apache.org/docs/1.8.4/manual/opennlp.html`. Accessed: 2018-02-28.

[Ost15] D. A. Ostrowski. Using latent dirichlet allocation for topic modelling in twitter. In *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, pages 493–497, Feb 2015.

[Ott17] Mark Otto. Bootstrap 4. `https://blog.getbootstrap.com/`, Dec 2017. Accessed: 2018-02-28.

[Ros18] Steve Rose. Ripe for a kicking: Hollywood's love-hate relationship with rotten tomatoes. `https://www.theguardian.com/film/2018/feb/26/rotten-tomatoes-hollywood-love-hate-relationship`, Feb 2018.

[SG16] Jasmeet Singh and Vishal Gupta. Text stemming: Approaches, applications, and challenges. *ACM Comput. Surv.*, 49(3):45:1–45:46, September 2016.

[Sid18] Sabrina Siddiqui. Trump renews attack on florida deputy: 'i'd run in there even if i didn't have a weapon'. `https://www.theguardian.com/us-news/2018/feb/26/gun-control-laws-nra-congress-return-recess`, Feb 2018.

[SMNR16] M. Sun, P. Mi, C. North, and N. Ramakrishnan. Biset: Semantic edge bundling with biclusters for sensemaking. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):310–319, Jan 2016.

[SOK$^+$16] H. Strobelt, D. Oelke, B. C. Kwon, T. Schreck, and H. Pfister. Guidelines for effective usage of text highlighting techniques. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):489–498, Jan 2016.

[SOR$^+$09] H. Strobelt, D. Oelke, C. Rohrdantz, A. Stoffel, D. A. Keim, and O. Deussen. Document cards: A top trumps visualization for documents. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1145–1152, Nov 2009.

[SR18] Kareem Shaheen and Andrew Roth. Syria: Putin orders five-hour daily ceasefires in eastern ghouta. `https://www.theguardian.com/world/2018/feb/26/syria-eastern-ghouta-assad-forces-un-ceasefire`, Feb 2018.

[VH17] Kirill Voronov and Gevorg Harutyunyan. Modularadmin. `https://github.com/modularcode/modular-admin-html`, Dec 2017. Accessed: 2018-02-28.

[Wal06] Hanna M. Wallach. Topic modeling: Beyond bag-of-words. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 977–984, New York, NY, USA, 2006. ACM.

Appendix

The survey for user feedback contained three documents with three pipeline results each. The input documents and results were the following.

Document 1

Document 1 is the following news article from The Guardian [Ros18]:

> Twenty years ago, the internet was a very different place. Google was a fresh rival to Alta Vista and Lycos. Apple computers looked like boiled sweets, and we dialled up to "surf the net", having installed the software via CD-Rom. The movie world of 1998 was also somewhat different: the box office was ruled by meteorite movies and Adam Sandler; Harvey Weinstein was an Oscar winner; and The Avengers was a lame, retro spy comedy with Ralph Fiennes and Uma Thurman. It was into this climate that Senh Duong launched Rotten Tomatoes – known in the business as RT – a site that has transformed both worlds, although nobody seems quite sure if it has done so for better or worse. Duong's idea was simple – to compile movie reviews – and it still drives Rotten Tomatoes. He was inspired by his love of Jackie Chan and Jet Li movies and would scour the internet looking for reviews of them. So why not put them in one place? Duong already had a full-time job, he says. "Rotten Tomatoes was a side project I worked on in the evenings." He single-handedly designed and coded the site in just two weeks. "It was very laborious. Every page was manually assembled using HTML. Every review was manually searched for, read and quoted." In the same way that, say, lastminute.com and Expedia compare plane ticket prices, Rotten Tomatoes' review aggregation has turned out to be super-useful, particularly as it boils all those reviews down to a single, convenient percentage score. It then boils down that score even further, to a simple graphic of a tomato. In the same way that Siskel and Ebert gave a "thumbs up" or a "thumbs down", or the man from Del Monte tasted a pineapple and said "yes" or "no", so Rotten Tomatoes' "Tomatometer" separates movies into "fresh" or "rotten". If at least 60% of a movie's reviews are positive, it is graded "fresh", signified by a ripe, red tomato. Less than 60% and it is "rotten", signified by a green splat.

Over 75% gets you a “certified fresh” logo, like a sticker on a quality piece of fruit. (The 1998 Avengers movie, if you were wondering, scored a supremely rotten 5Today, movies supposedly live or die by the ripeness of that virtual fruit. Rotten Tomatoes has become the one movie site to aggregate them all. The Tomatometer appears not only on Rotten Tomatoes’ site but also on ticketing sites such as AMC cinemas and Fandango (which has owned Rotten Tomatoes since 2016). It comes up on Google searches, iTunes, SoundCloud, in Twitter and chatroom discussions and (as long as the rating is “fresh”) in movie studios’ marketing campaigns. It is a news item when a movie achieves a “100% fresh” rating, as recently happened with Paddington 2 and, before that, Greta Gerwig’s Lady Bird. With its dominance and prominence, Rotten Tomatoes is becoming the story – and not always in a good way. After Lady Bird got its 100% score, for example, one critic opted to lob a green splat into the mix, not because he hated the movie, but because everyone else liked it so much. “I had to consider whether to cast Lady Bird as fresh or rotten in the context of a perfect score that people were using to trumpet Lady Bird as the all-time best-reviewed movie on RT,” Cole Smithey tweeted. In other words, Rotten Tomatoes’ status as a neutral measure of critics’ opinions comes into question when it starts to influence those opinions. The possible gaming of Rotten Tomatoes scores has taken on more sinister aspects lately. Earlier this month, Facebook announced it had taken down the page of a group called Down With Disney’s Treatment of Franchises and Its Fanboys, which was attempting to orchestrate a mass troll assault on the Rotten Tomatoes score of the superhero movie Black Panther. Alongside the critic-designated Tomatometer score, Rotten Tomatoes also gives each movie an “audience score”, determined by registered users and represented by a popcorn bucket: red and full for positive; green and tipped-over for negative. The anti-Black Panther group sought to lower the movie’s audience score by bombarding the site with negative reviews. It claimed to have programmed bots to create fake user accounts. It also said it was acting in the name of DC comics, the main rival to Black Panther’s (Disney-owned) Marvel, but suspicions of far-right motivations persist, particularly because the same group had previously targeted Star Wars: The Last Jedi (also Disney-owned) on account of its supposed “social justice warrior concepts”. Rotten Tomatoes has denied the attacks succeeded, but at present The Last Jedi’s Tomatometer score is 91% (a critical “Yay!”) while its audience score is 48% (a public “Meh”). Was this discrepancy the result of far-right bots or genuine audience division? Either way, it didn’t matter much: The Last Jedi is now the ninth-highest-grossing movie in history. Black Panther is likely to be a billion-dollar movie, too. When movies bomb, however, the studios have been quick to blame Rotten Tomatoes. Last summer, Hollywood resorted to tomato-shaming to spare its own blushes over colossal failures such as Baywatch (Tomatometer score: 18%), The Mummy (16%), King Arthur: Legend of the Sword (29%) and

Pirates of the Caribbean 5 (30%). “The critic aggregation site increasingly is slowing down the potential business of popcorn movies,” complained the website Deadline. Director Brett Ratner called Rotten Tomatoes “the worst thing we have in today’s movie culture” and “the destruction of our business”. He may have been stung by the fate of Warner Bros’ blockbuster Batman v Superman: Dawn of Justice, which Ratner’s company co-produced; it earned a malodorous 27%. The situation came to the boil with Batman v Superman’s 2017 follow-up: Justice League. For Warner Bros, the movie was a big deal: a superhero team-up with an estimated $300m budget. So, eyebrows were raised when Justice League’s Rotten Tomatoes score did not appear on the site as expected, once an embargo on critics’ reviews lifted. Even when those reviews were available on other sites and the movie was previewing in cinemas, Rotten Tomatoes’ webpage for Justice League was blank. Instead, the excuse ran, Justice League’s score was to be announced on Rotten Tomatoes’ new web show, See It Or Skip It, in which presenters provide “context and conversation” around the movie of the week before revealing its all-important Tomatometer score. For Justice League, that score was a decidedly unripe 43%. By the time it appeared on the website, it had dropped to 40%. Some observers smelled a conspiracy, since Warner Bros holds a 30% stake in Rotten Tomatoes’ parent company, Fandango (Universal owns the other 70%). Rotten Tomatoes, however, denied Warner Bros had anything to do with the decision: “We are absolutely autonomous, like any news organisation,” it said. “There is no outside influence on anything we put on the site.” If the studio was secretly trying to bury bad news, it didn’t work. The incident ultimately generated negative publicity for Justice League, Warner Bros and Rotten Tomatoes.

The results are displayed in Figures 1, 2, and 3.

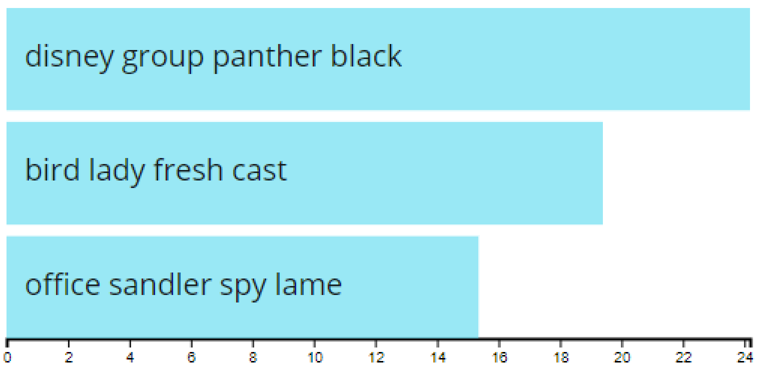


Figure 1: Local topic modeling result for document 1 using pipeline 1.

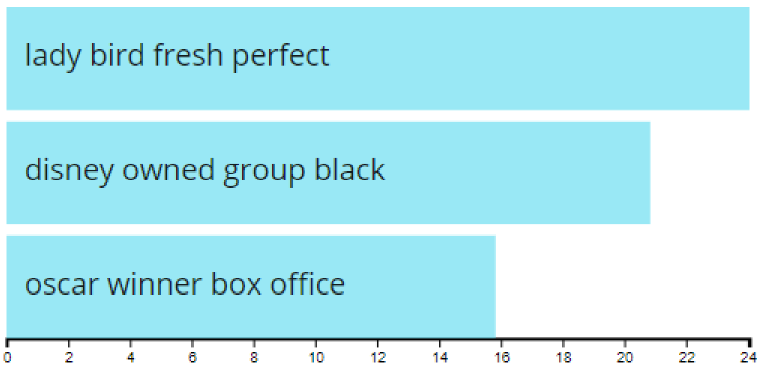


Figure 2: Local topic modeling result for document 1 using pipeline 2.

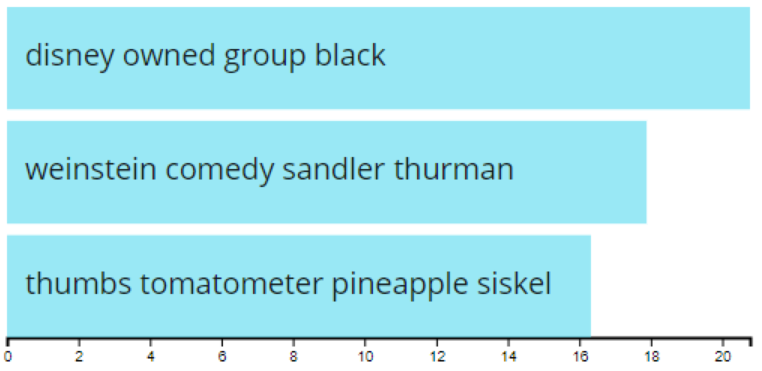


Figure 3: Local topic modeling result for document 1 using pipeline 3.

Document 2

Document 2 is also a news article from The Guardian [SR18]:

> Vladimir Putin has ordered a daily five-hour “humanitarian pause” in the besieged Syrian enclave of eastern Ghouta to begin on Tuesday, effectively replacing a United Nations security council resolution that had demanded a month-long ceasefire in the embattled region. The Russian president’s move, which was announced by his defence minister, Sergei Shoigu, highlighted in stark terms Russia’s primacy in Syrian affairs and the UN’s failure to impose an end to the fighting in the area bordering Damascus. More than 500 people have been killed in eight days of one of the deadliest bombing campaigns by the regime of Bashar al-Assad and his allies during the seven-year war. The move by Moscow follows mounting condemnation of the violence, with the UN secretary general, António Guterres, describing the situation in Ghouta as “hell on earth”. Shoigu said a ceasefire would begin on Tuesday in the Damascus

suburb and would take place from 9am until 2pm (7am GMT to 12pm) daily, according to a transcript of his remarks published by the Russian news agency Interfax. Russia, a key ally of the Syrian regime, would also help create an evacuation route for civilians in the area, he added. The Syrian Observatory for Human Rights said calm had generally prevailed in eastern Ghouta since midnight, though four rockets had hit the town of Douma on Tuesday morning. The announcement on Monday came after at least 29 people were killed in eastern Ghouta despite a UN security council resolutionthat demanded an immediate cessation of the fighting. Local doctors and monitors said at least 18 people had been injured by a suspected chlorine attack in eastern Ghouta on Sunday evening. Residents have condemned the international community's inability to put an end to the fighting. "I am embarrassed for the UN security council," said Ghanem Tayara, the chairman of the Union of Medical Care and Relief Organisations, which helps run dozens of hospitals in Syria. "The mightiest nations on the planet cannot enforce the most basic standards of human rights and decency." The latest deaths in eastern Ghouta brought the weeklong carnage in the enclave to more than 500 killed in airstrikes and shelling by forces loyal to Assad. The security council resolution, which was unanimously approved on Saturday, called for a monthlong ceasefire "without delay". "Eastern Ghouta cannot wait. It is high time to stop this hell on earth," Guterres told the UN Human Rights Council in Geneva. Zeid Ra'ad al-Hussein, the high commissioner for human rights, condemned the security council's failure to end seven years of "unremitting and frightful mass killing". He said: "Second to those who are criminally responsible – those who kill and those who maim – the responsibility for the continuation of so much pain lies with the five permanent members of the UN security council. So long as the veto is used by them to block any unity of action, when it is needed the most, when it could reduce the extreme suffering of innocent people, then it is they – the permanent members – who must answer before the victims." The violence has highlighted the Syrian government's desire, alongside its allies in Moscow and Tehran, to score a military victory in the area, which has been under a tightening siege for nearly a year and is strategically significant owing to its proximity to Damascus. Reports of an attack using chlorine, which has been used frequently in the past by the Assad regime, would represent a further escalation if confirmed. Doctors said they had treated patients in the town of Shifounieh after a bombing. The patients exhibited symptoms consistent with exposure to chlorine, including respiratory problems, inflammation in the eyes and mucous membranes, as well as cases of hysteria and dizziness, the doctors said. A video released by the Syrian American Medical Society, which also helps run hospitals in opposition-held parts of Syria, showed children and first responders doused with water and breathing through oxygen masks. "The most heartbreaking thing is the children and infants who are brought into the hospital from under the rubble and their entire family has been killed

– mother, father and siblings," said one doctor in Ghouta. "Where do we go with this child? Children whose age is in the single digits who have known nothing except fear and terror and death and shelling. Babies and children treated in Ghouta hospital on Sunday "The humanitarian organisations have failed. I wonder if we can appeal to the animal rights organisations," he added. The watered-down resolution passed by the security council did not have a timeframe, though it called for a ceasefire "without delay" and the lifting of sieges as well as the delivery of humanitarian aid. On Monday, the French president, Emmanuel Macron, told his Turkish counterpart, Reçep Tayyip Erdoğan, that the UN's proposed 30-day ceasefire must be applied across the country, including in Afrin, where Turkey is waging an offensive against a Kurdish militia. It "must be put into effect everywhere and by everyone without delay", Macron said in a telephone call to Erdoğan, adding that Turkey, Russia and Iran, the three countries overseeing talks in Astana aimed at ending the civil war, "have a direct responsibility in this regard that must be applied on the ground". Ankara launched an offensive last month against the Kurdish People's Protection Units (YPG) in Afrin in northern Syria. On Sunday the Turkish government said the UN ceasefire would not affect its operation, which it claims is aimed at fighting "terrorist organisations that threaten the territorial integrity and political unity of Syria". Turkey sees the YPG as the Syrian branch of the Kurdistan Workers' party (PKK), which for more than three decades has waged an insurgency against the Turkish state and is classified by Turkey, the US and the European Union as a terrorist group. But the offensive has raised tensions with Washington, which works closely with the YPG in the fight against jihadists in Syria.

The results are displayed in Figures 4, 5, and 6.

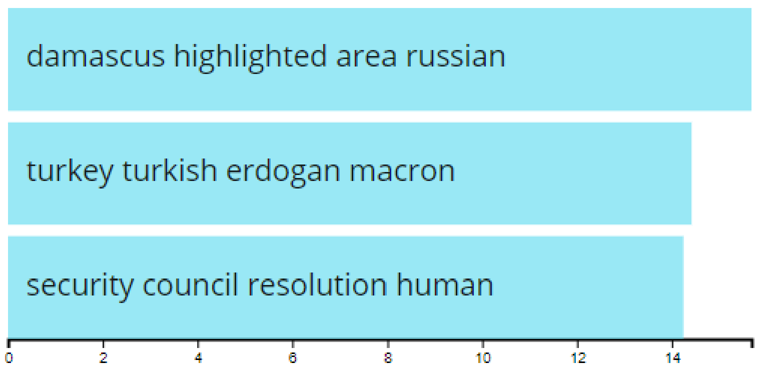


Figure 4: Local topic modeling result for document 2 using pipeline 1.

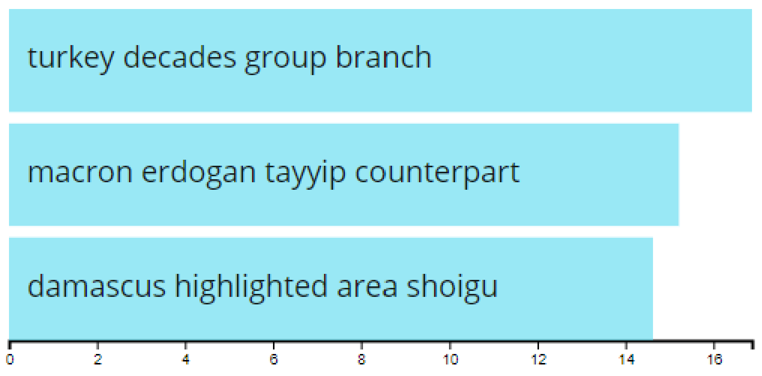


Figure 5: Local topic modeling result for document 2 using pipeline 2.

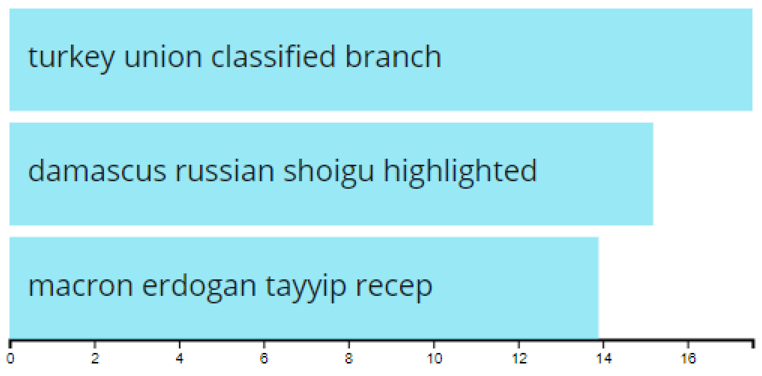


Figure 6: Local topic modeling result for document 2 using pipeline 3.

Document 3

Document 3 is another news article from The Guardian [Sid18]:

> Addressing a gathering of 39 state governors at the White House, Trump said officers who were outside the school at the time of the shooting “weren’t exactly medal of honor winners”. “The way they performed was really a disgrace,” he added. “I really believe I’d run in there even if I didn’t have a weapon.” The president spoke as lawmakers returned to Washington following a week-long recess and amid intensifying debate over gun laws. The 14 February massacre at Marjory Stoneman Douglas high school, in which 17 people were killed, has forced Congress to contend yet again with one of the most politically controversial issues. Trump has pushed for arming teachers – a proposal that has been pilloried by educators. In doing so he has repeatedly criticised Scot Peterson, the armed school resource deputy who waited outside Marjory Stoneman Douglas high school as the shooting transpired. Peterson resigned, after being suspended without pay. Trump

has publicly attacked Peterson as a “coward” who he said “doesn’t love the children, probably doesn’t know the children”. Peterson denied the allegation in a written statement released through his lawyer on Monday, saying he had not entered the building because he believed the gunfire was coming from outside. “Mr Peterson wishes that he could have prevented the untimely passing of the 17 victims,” the attorney, Joseph DiRuzzo, wrote. “However, the allegations that Mr Peterson was a coward and that his performance, under the circumstances, failed to meet the standards of police officers, are patently untrue.” On Sunday, the Broward County sheriff, Scott Israel, told CNN he was investigating the behavior of three other deputies, who the network reported had been at the campus but had not entered the school. Moment of silence for Florida school shooting victims on House floor – video On Monday, Trump continued to call on Congress to take action on gun law – a departure from his response to previous shootings under his watch. The president has signaled support for tightening background checks and instructed the Bureau of Alcohol, Tobacco, Firearms and Explosives to work on a memorandum to outlaw bump stocks – the attachments that enable semiautomatic rifles and other devices to fire faster. Lawmakers are likely, however, to face familiar obstacles in passing even modest legislation, as midterm elections loom. The National Rifle Association has also come out aggressively against any new restrictions, even as public support for stricter gun laws has risen to its highest level since the early 1990s, according to a CNN poll released on Sunday. Trump told the group of governors on Monday he believed the NRA was open to at least some changes to gun laws, noting he had had dinner with the group’s leaders Wayne LaPierre and Chris Cox over the weekend. “Don’t worry about the NRA, they’re on our side,” Trump said. “Half of you are so afraid of the NRA. There’s nothing to be afraid of.” Although mass shootings had become almost routine in the US, drawing little reaction in Washington, events in Florida have spawned a rare grassroots backlash, led by students. The national outcry in the wake of Parkland could tee up the first major debate over gun laws since the 2012 massacre at the Sandy Hook elementary school in Newtown, Connecticut, in which 20 young children and six adults were killed. The Senate failed to expand background checks in the aftermath of Sandy Hook, due to a Republican-led filibuster of a bipartisan bill joined by a handful of Democrats. But several Democrat and Republican senators who were in office then said on Monday that the ground “has shifted”, owing in large part to the vocal activism of Parkland student survivors. Several pieces of legislation are under consideration. But only incremental measures appear to have any chance of passing, including a bipartisan proposal aimed at ensuring that states and federal agencies comply with and accurately report criminal and mental health records to the National Instant Criminal Background Check System (Nics). The White House has said Trump supports the bill, known as Fix Nics, but may seek some revisions to its language. 'We must immediately harden our schools' says NRA's Wayne LaPierre – video A pair of senators are also poised to introduce bipartisan legislation that would raise the age for buying assault weapons, including the AR-15 used by the gunman in Parkland, from 18 to 21. Trump was supportive of the idea in the initial days after the Parkland shooting but has been less vocal since the NRA came out against the proposal last week. Sarah Sanders, the White House press secretary, told reporters on Monday that Trump was "still supportive of the concept" and denied the president had been influenced by the NRA. The president has nonetheless sent mixed signals on the issue of guns, most often returning to the controversial suggestion that the way to prevent school shootings such as that in Florida is to arm teachers. The proposal, which is backed by the NRA, is unlikely to garner support on Capitol Hill – but is indicative of Trump's allegiance to gun rights activists and his base. Trump signaled on Monday he was open to taking on the NRA if necessary, stating: "If they're not with you, we have to fight them every once in a while. "They're doing what they think is right," he added. "But sometimes we're gonna have to be very tough and we're gonna have to fight them."

The results are displayed in Figures 9, 7, and 8.

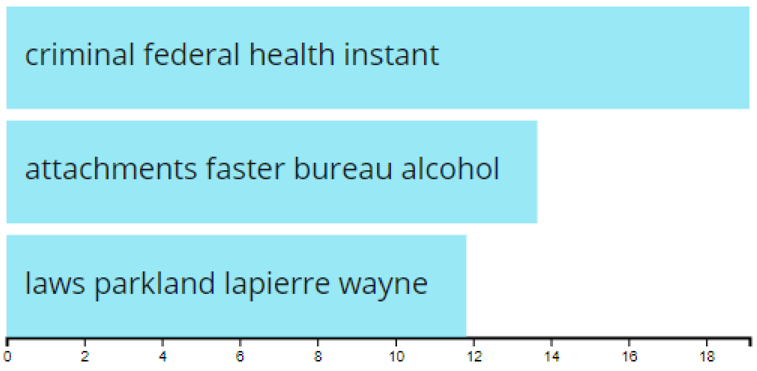


Figure 7: Local topic modeling result for document 3 using pipeline 1.

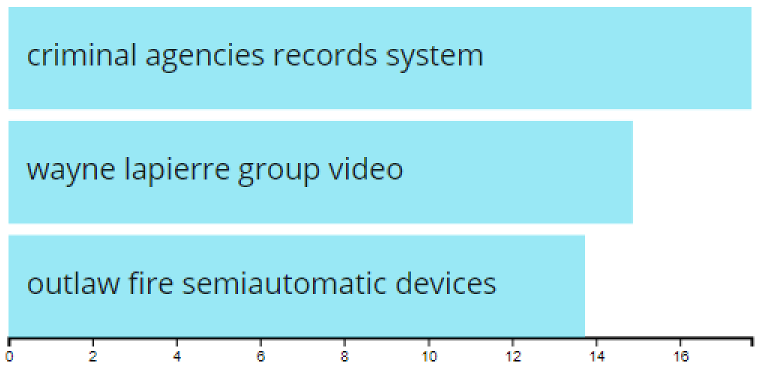


Figure 8: Local topic modeling result for document 3 using pipeline 2.

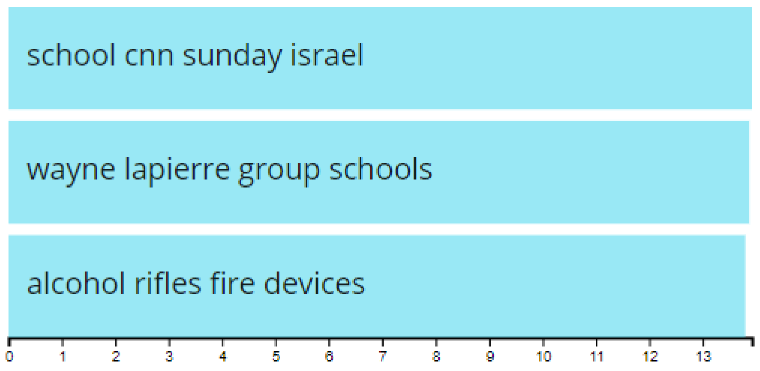


Figure 9: Local topic modeling result for document 3 using pipeline 3.