KeywordScape: Visual Document Exploration using Contextualized Keyword Embeddings

Henrik Voigt[1], Monique Meuschke[3], Sina Zarrieß[2] and Kai Lawonn[1]
[1]University of Jena [2]University of Bielefeld [3]University of Magdeburg
[1]first.last@uni-jena.de
[2]first.last@uni-bielefeld.de
[3]last@isg.cs.uni-magdeburg.de

Abstract

Although contextualized word embeddings have led to great improvements in automatic language understanding, their potential for practical applications in document exploration and visualization has been little explored. Common visualization techniques used for, e.g., model analysis usually provide simple scatter plots of token-level embeddings that do not provide insight into their contextual use. In this work, we propose KeywordScape, a visual exploration tool that allows users to overview, summarize, and explore the semantic content of documents based on their keywords. While existing keyword-based exploration tools assume that keywords have static meanings, our tool represents keywords in terms of their contextualized embeddings. Our application visualizes these embeddings in a semantic landscape that represents keywords as islands on a spherical map. This keeps keywords with similar context close to each other, allowing for a more precise search and comparison of documents.

1 Introduction

Recent work in Natural Language Processing (NLP) has brought great advances in the contextual modeling of word meanings in texts (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2019). When it comes to visualizing the meaning contained in a collection of documents, keywords still play an important role (El-Assady et al., 2020; Ji et al., 2017; Kim et al., 2017). Keyword-based methods have a number of advantages because they are intuitive and easy to visualize, for example, in a bar chart showing the frequency of a keyword over different years in a document collection. However, one of the major limitations in existing approaches is that it assumes that a keyword has a static meaning across different texts and domains. This assumption is highly unrealistic (Schütze, 1998; Navigli, 2009). A term like *training* means something quite different in the context of *machine learning* than in *psychology*.

There are tools for visualizing document collections at different levels of granularity. To get an overview of a set of texts, it is often beneficial to use visualizations that group texts on a high level according to their overarching meaning, also called *topic*. This is done by John et al. (2019); Kim et al. (2017); Dang and Nguyen (2018); Le and Akoglu (2019), e.g. to distinguish scientific texts on the topic of *machine learning* visually from texts from the field of *psychology*. To find documents that match a semantic query, e.g., a list of related keywords such as *learning, curriculum, pre-training*, low-level document exploration is required. Here, users must evaluate the meaning of specific paragraphs, sentences, and keywords in a given context. Current visual document exploration systems do not use contextualized neural representations and rely on topic models or frequency-based keyword clustering techniques (Wang et al., 2014; Ganesan et al., 2015; Kim et al., 2017; Yang et al., 2017; Ji et al., 2019; John et al., 2019). These allow high-level comparison of documents but are unable to distinguish content based on low-level semantic meaning.

In this work, we propose KeywordScape, a tool that visualizes keywords in their semantic contexts as islands on a map to support meaning-driven document exploration. This makes it possible to explore the potential of contextualized word embeddings for visual document exploration by exploiting their strengths for disambiguation.

In the following, we clarify how our work fits into the current research context. We demonstrate the applicability of contextualized keywords in a visualization system architecture, explain the user interactions it supports, and show its application in use cases that solve real-world problems. The main contributions of this paper include:

- Provision of a novel method for visualizing

Proceedings of the The 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 137 - 147
December 7-11, 2022

contextualized keyword embeddings as visual islands.

- Design and implementation of a meaning-preserving visual document exploration system.

2 Related Work

Visualization of text has been explored extensively in the VIS community. This section refers to related work in the sub-field of visual document exploration, reviewing relevant methods and ideas for visualizing document collections. Furthermore, we show how recent developments in the field of NLP, and in particular neural word embeddings, feed into these solutions.

2.1 Visual Document Exploration

Visual exploration of documents is a well-researched task (Heimerl et al., 2016; Mitra and Craswell, 2017; Zhang et al., 2018; John et al., 2018; Han et al., 2018). The most frequently studied methods can be categorized into visual topic modeling (Dou et al., 2013; Kucher et al., 2018a; John et al., 2019; Kim et al., 2017; Dang and Nguyen, 2018; Le and Akoglu, 2019), visual information retrieval (Koch et al., 2014; Kraker et al., 2016; Heimerl et al., 2016; Dias et al., 2019) and visual sentiment analysis (Dai and Prout, 2016; Martins et al., 2017; Kucher et al., 2018b, 2020). Visual topic modeling is most closely related to our approach. A topic model generally aims to extract groups of keywords as coherent topics and assign the documents in the collection to these topics. Visual topic modeling supports the exploration of these topic models by visually representing the extracted topics and making them interactive. A large proportion of current applications for visual topic modeling such as VISTopic (Yang et al., 2017), LDAExplore (Ganesan et al., 2015) or TopEx (Olex et al., 2021) in their NLP pipelines is based on methods like LDA (Blei et al., 2003), LSA (Deerwester et al., 1990) or HDP (Wang et al., 2011). The visualization pipeline relies on clustering algorithms like K-Means (Kanungo et al., 2002) or dimensionality reduction methods like PCA (F.R.S.), t-SNE (Maaten and Hinton, 2008), or UMAP (McInnes et al., 2018).

2.2 Neural Embedding Visualization

Visualizations of static neural word embeddings are used to explore document spaces (Berger et al., 2017; Ji et al., 2017, 2019) sentiment spaces (Dai and Prout, 2016; Martins et al., 2017; Kucher et al., 2020) or concept spaces (Park et al., 2018; Heimerl and Gleicher, 2018). Depending on how the visual models are constructed, the approaches can be divided into neural embedding visualizations (Mitra and Craswell, 2017; Chen et al., 2018; Li et al., 2018) and interactive human-in-the-loop applications (El-Assady et al., 2020; Park et al., 2018). In both, the idea of representing individual word tokens in a condensed form that captures their semantic meaning is applied. The high-dimensional representations are reduced to a lower dimension and visualized with colored dots or icons as visual metaphors for individual words or documents (Smilkov et al., 2016; El-Assady et al., 2020). Complete static embedding spaces are visualized in Li et al. (2016); Chen et al. (2018); Molino et al. (2019). Liu et al. (2018) visually investigate the semantic relations in embedding spaces of static word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) embeddings. Our work differs from this in that, unlike in the methods presented above, the contextual embedding spaces are not a fixed set and dynamically new embeddings are created for each token in the respective document based on its occurrence in the context.

3 KeywordScape System

We propose a system architecture shown in Figure 1 consisting of a **NLP Pipeline** and a **Visualization Pipeline**. The first parses the documents into a map of contextualized representations based on BERT (Devlin et al., 2019) and UMAP (McInnes et al., 2018). The second is responsible for the visualization and is based on the D3 library (Bostock et al., 2011), the HTML canvas element for rendering, and SVG for additional data manipulation and information display. For a video demonstration of the system, visit `https://youtu.be/6jaF7HiPzTk`.

3.1 NLP Pipeline

3.1.1 Text Processing

Parsing. Each document of the collected set is parsed into a text string. The free library science-parse from AllenAI (sci) is used. For each document *title*, the *authors*, the *year* of publication, the *abstract*, and the *text* of the document are extracted.
Cleaning. After parsing the documents into text strings, the text is broken down into word tokens

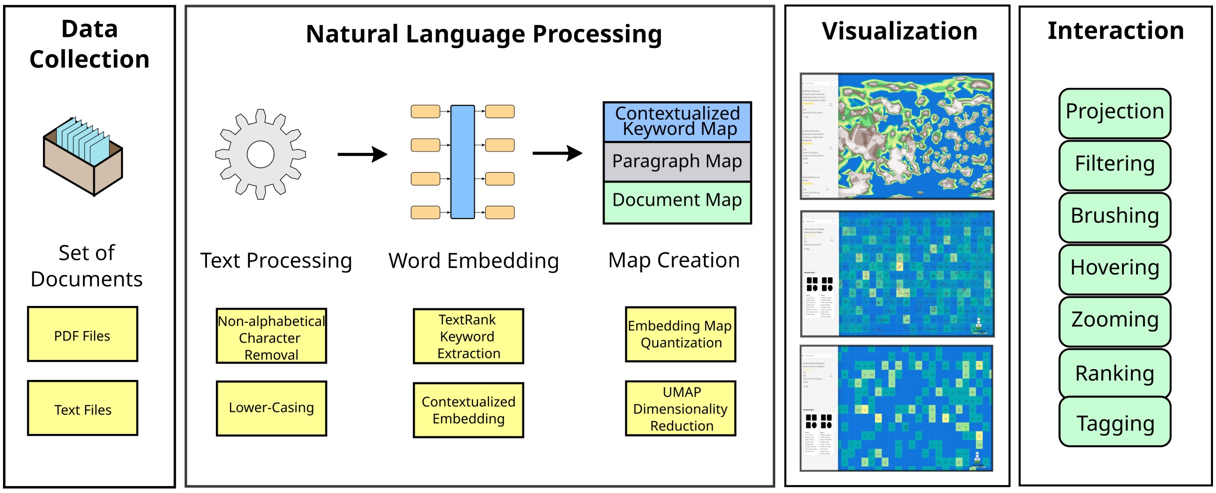

Figure 1: The KeywordScape System Architecture. Within the **NLP Pipeline**, the document collection is preprocessed, keywords are extracted and embedded alongside documents and paragraphs. The high-dimensional representations are projected onto a unit sphere and forwarded to the **Visualization Pipeline**. Here, maps of different granularity can be switched and explored with a variety of interaction techniques.

using the SpaCy library (spa). A cleaning procedure is applied to filter out all non-lexical tokens.

Paragraphing. The document is divided into paragraphs with a size of 512 tokens or less. This corresponds to the encodable context width of the BERT model (Devlin et al., 2019).

Keyword Extraction. A set of N different keywords is determined by a user-selected extraction method, either TextRank (Mihalcea and Tarau, 2004), RAKE (Rose et al., 2010), or TF-IDF (Ramos, 2003). A single document is thus visually represented by the semantic coverage of its most relevant N contextualized keywords. In order to be consistent across the collection and to adapt to the respective document size, the number of keywords is calculated individually for each document. To achieve this, the user determines what percentage $p\%$ of words in a document are treated as keywords. For our visualizations, we set the keyword percentage p per document to 5%.

3.1.2 Word Embedding

For each document, each paragraph is passed to the BERT model and the embedding vector of the last layer of all tokens with dimension 768 is extracted. All word embeddings that do not belong to keywords are removed. If a keyword is composed of subword embeddings, the average embedding is used. This results in a contextualized representation of each keyword based on the paragraph in which it occurs. Unlike methods with pure keyword extraction, this provides a fine-grained representation of meaning. Each of the contextualized keyword embeddings is labeled by its lemmatized word form to reduce unnecessary variance and facilitate navigation in the visualization.

3.1.3 Map Creation

Map Granularity. Three maps are created to enable iterative exploration of document collections at different levels of semantic granularity. The first map is a **document map** that creates a single representation for each document by using a SentenceBERT embedding of its abstract (Reimers and Gurevych, 2019). The second map is a **paragraph map** that applies the same technique to all paragraphs in the corpus. The third map is a **contextualized keyword map**, which is the focus of this work and will therefore be explained in more detail in the following subsections.

Dimensional Reduction. The contextually embedded keyword vectors are reduced to a lower, plottable dimension using the UMAP algorithm (McInnes et al., 2018; uma). A unit sphere is used as the reduction space, which allows points to be treated as [lat, lon] expressions. Reducing the points to the surface of a unit sphere has the advantage that all embeddings are mapped relative to all other embeddings on the sphere. This offers a smooth visual exploration of the keyword landscape.

Quantization. To avoid overlapping points in the visual map and to make the number of points manageable for the browser-internal visualization pipeline, the unit sphere is quantized into elementary quadrilaterals, each of which represents all

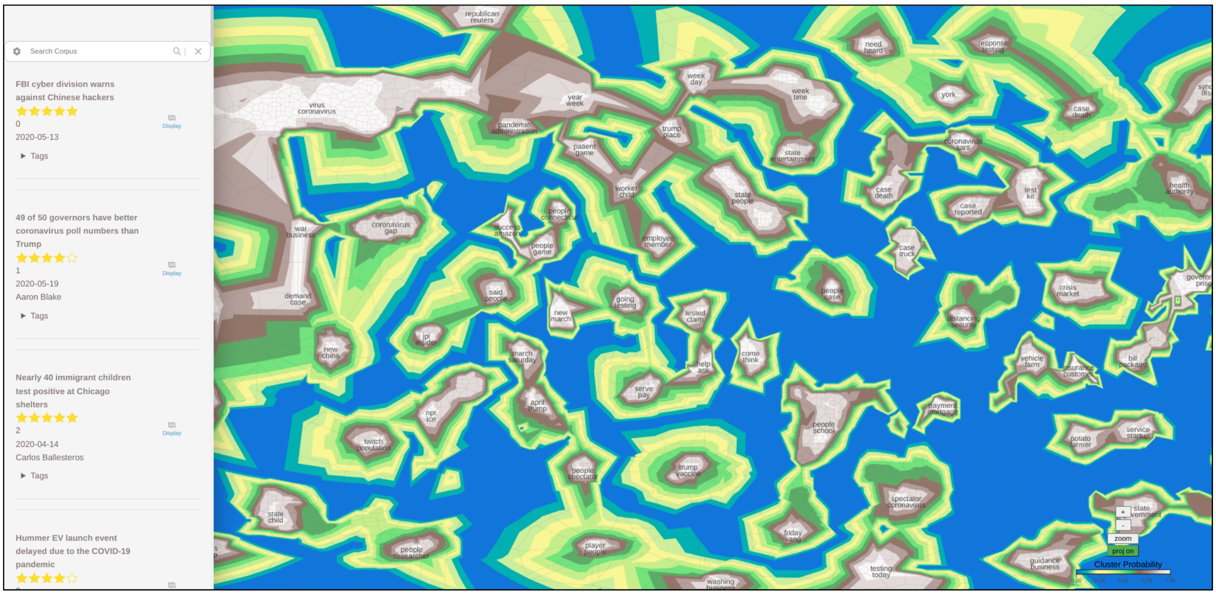

Figure 2: Overview of the KeywordScape interface. *Left:* Search interface that allows users to search for documents in the collection. *Right:* Map visualization of keyword islands of the ReCOVery corpus with interaction bar.

points contained in its area. The minimum side length of a quadrilateral is 0.1 degrees. With the help of this quantization, the approach can be scaled up to 250 documents using a keyword percentage of 5%. This ensures a fast construction of the map in the browser and enables fluid interaction.

3.2 Visualization Pipeline

3.2.1 Document Map and Paragraph Map

The document map and paragraph map were created with the aim of allowing users to iteratively refine the granularity of the context of interest during visual exploration. The document map provides an overview of the collection. Each document is represented as a single point on the unit sphere that positions semantically similar document embeddings close to each other. As a visual metaphor for a coherent semantic region, labeled and coloured grid cells are used as an adaptation of the visualization technique *heatmap*. The unit sphere is divided into a grid of user-defined size. For each grid cell, the most frequently occurring keywords in it are displayed adaptive to the zoom level. The cells are coloured according to the density of dots in each cell. The higher the density of dots in a cell, the lighter the colour. In this way, document clusters can be quickly identified, and at the same time, one gets an idea of the most important keywords in these clusters. The paragraph map shows the distribution of paragraph embeddings over the entire unit sphere. Similar to the document map, a coloured grid with adaptive labeling is used. The map makes it possible to find topics covered in individual paragraphs, allowing for a more detailed examination than at the document level.

3.2.2 Contextualized Keyword Map

The contextualized keyword map (see Figure 2) is the focus of this work and enables a visual keyword search that takes into account the semantic meaning of the individual keywords. The embedding points in [lat, lon] of the contextualized keyword map are projected onto a *geoequirectangular* map projection. The geo-voronoi microlibrary (geo) of D3 is used to create a Delaunay triangulation of the points on the sphere. A Voronoi map is calculated from this triangulation. Each Voronoi region represents a set of quantized word points in the space of the semantic map. The visual metaphor of the sea and *visual islands* is used to represent a keyword context. Islands represent clusters of keywords that occur together in the same context. To determine the visual islands, an HDBSCAN (McInnes et al., 2017) clustering is applied to the Voronoi map, which assigns a cluster probability to each Voronoi region with respect to the cluster probability of its center. Cluster probability is the basis for colour coding, in that higher probabilities encode a darker, land-like colour and lower probabilities encode a blue, sea-like colour, resulting in the creation of *visual islands*. The positions of the islands in relation to each other on the map represent their semantic

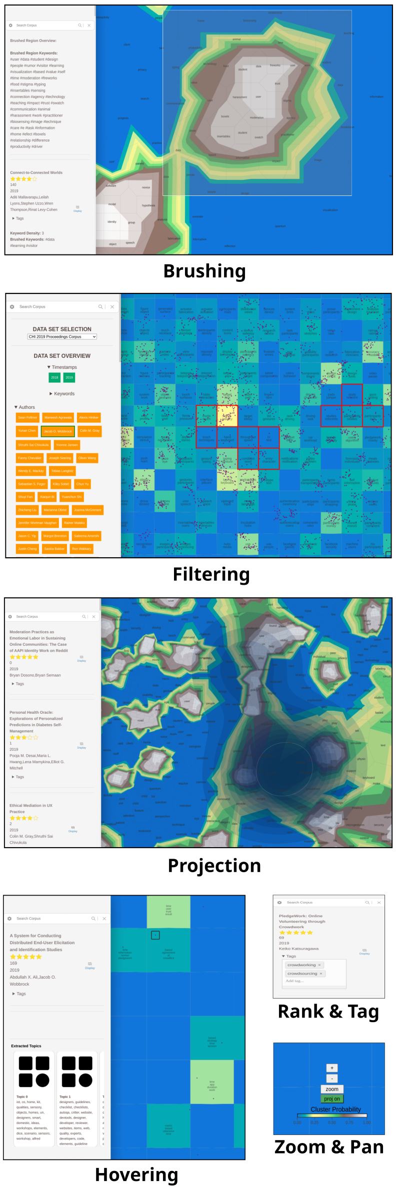

Figure 3: The main interaction techniques *projection, filtering, brushing, hovering, rank & tag, zoom & pan* within the KeywordScape interaction loop.

distance. The labels of the keywords in a visual island represent the keywords with the highest frequency in the respective Voronoi region.

4 User Interaction

KeywordScape offers a number of interaction techniques, which are illustrated in Figure 3 and explained below:

Projections. To create a projection a search query in form of a set of keywords is translated into a selection of embedding points. From the resulting set of points, a contour map is interpolated based on the density of the selected points in each region of the map. The contour map is then projected onto the contextualized keyword map. This makes it possible to visualize a user's search similar to rain showers on a weather map, as shown in Figure 3.

Filtering. Filtering is used in combination with the projection function. The user can define queries to the system either via the text-based search or via predefined query selectors such as *timestamps, keywords, authors, star ratings*, or *tags*. The query is then translated into a projection condition and a visual expression of the query is projected onto the map so that the user can visually understand where within the semantic landscape of the document collection the query applies. For example, if the user selects a particular author as a query selector, all documents that the author has contributed to the corpus are filtered out, their keywords extracted and projected onto the KeywordScape. With the help of this view, the user can easily see which semantic regions the author mainly focuses on.

Zooming. A filtered semantic map generates regions of interest that can be examined in detail using the zoom function. Zooming in combination with adaptive region labeling makes it possible to examine the keywords of a context island with increasing granularity.

Brushing. Brushing highlights a specific area of interest to find the documents with keywords in that area. Visual islands characterized by a collection of keywords occurring in the same context are made visually tangible in this way and the documents containing these contexts are listed. An application of the brushing feature is shown in Figure 3.

Ranking. Ranking allows the corpus to be adapted to personal interests. Users award stars in a range from 0-5, making it possible to quickly filter out personally relevant documents, e.g. as support when writing a survey paper.

Tagging. Assigning tags to the documents allows quick access to document sets. This can be combined with the projection function to visualize the map coverage of several selected documents. In addition, tagging and projection are used to visually compare documents by dividing them into different tag groups and projecting the similarities as well as the differences of the two tag groups against each other, as shown in Figure 5.

Hovering. The hovering interaction is applied to the document map and the paragraph map. The tool displays the corresponding document in the search bar when the mouse pointer hovers over its representation in the map. When clicked, a de-

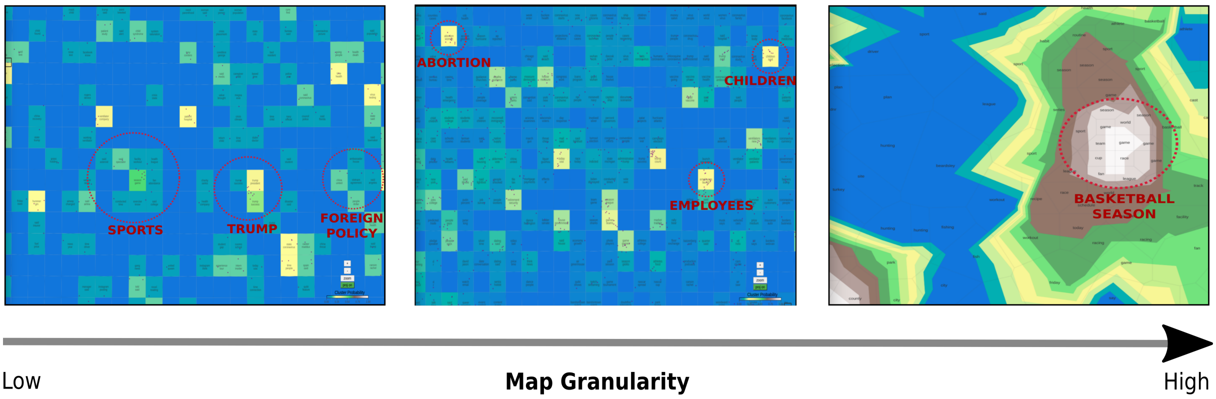

Figure 4: *Document, paragraph, and contextualized keyword map* of the ReCOVery corpus showing the semantic content of newspaper articles during the COVID-19 pandemic at different levels of granularity. Left: The *document map* allows users to identify documents with a similar main topic such as *sports, trump*, or *foreign policy*. Center: The *paragraph map* highlights more specific contexts below the document level, such as women suffering from *abortions* caused by COVID-19 infections, the situation of *employees*, and the impact of the COVID-19 pandemic on *children*. Right: The *contextualized keyword map* focuses on fine-grained keyword islands like the *basketball season*.

tailed document preview is displayed, containing meta-information such as *title*, *author*, *date* of publication as well as the most important *keywords* of the document and its *abstract*. The interaction techniques are evaluated in a user study. For a detailed description, please refer to the appendix A.

5 Use Cases

We apply the KeywordScape tool to two different document data sets. To show the applicability of the tool for everyday use, e.g., to get an overview of the content of news articles, we visualize the **reliable** sources of the *ReCOVery Corpus* of Zhou et al. (2020) in Section 5.1. These include newspaper articles from trusted news outlets. To illustrate the tool's ability to decompose the semantic occurrence of keywords into different contexts, we visualize the *food.com* data set from Majumder et al. (2019), which is used to generate personalized recipes from user preferences in Section 5.2.

5.1 Newspaper Articles

We visualize 250 newspaper articles from trusted news outlets of the ReCOVery corpus (Zhou et al., 2020). The document map provides an overview of the main topics covered in the articles, such as *sports, foreign policy*, or *trump* (see Figure 4). In particular, sources related to *trump* appear to have high coverage. At the paragraph level, it is particularly interesting that the context of *abortion* as a problem of women struggling with a COVID-19 infection is mentioned in many paragraphs. A user

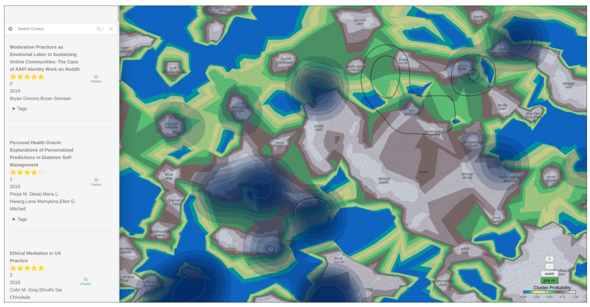

Figure 5: Visualization of *similarities* and *differences* of documents with regard to their keywords in context. Cividis coloured contours show regions where the contents of the documents overlap, black outlined contours show regions where the contents of the documents differ from each other.

with only a document-level visualization might not have been able to detect this because the contexts in which this subject is discussed are hidden behind the larger topic of the article. Using the contextualized keyword map, specific contexts in the semantic region of *sports* can be explored. For example, articles debating the progress of the *basketball season* can be found by brushing their contextualized keywords in the associated context island, as shown in Figure 4. The ability to examine keywords in relation to their meaning in context proves particularly useful for high-interest terms such as *trump*. A large number of articles use the polarizing keyword to attract interested users. A decomposition into the individual contexts in which it occurs, e.g. *vaccine* or *china*, would not be possible with a conventional keyword search, because the meaning of the word

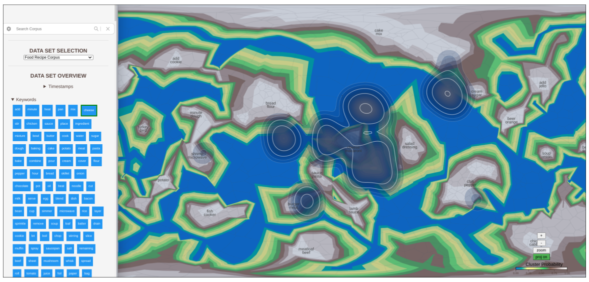

Figure 6: Contextualized keyword map of food recipes from the *food.com* data set. *Contours* outline the density of the keyword *cheese* within the different contexts of the map.

is not taken into account with these methods.

5.2 Food Recipes

We visualize 1500 recipes from the *food.com* data set (Majumder et al., 2019). Recipes, by their very nature, consist of several ingredients that are combined into one dish. An example question for recipe exploration could be: In which cooking contexts are certain ingredients used together? A conventional keyword search for the ingredient *cheese* would give us all the recipes that use cheese, no matter what the context. By projecting the keyword *cheese* onto the contextualized keyword map, as shown in Figure 6, a nice decomposition of the different cooking contexts in which cheese is used becomes visible. Obviously, *cheese* is very popularly used in a cooking context with *macaroni*, as in the dish *Mac'n'Cheese*. It also makes frequent appearances in *burger* recipes. Some *cheeses*, such as *cream cheese*, are used in the context of baking recipes. This shows the appearance of the keyword near the visual island of the main keyword *cake*. Others occur near the *bread* island in connection with the production of *sandwiches*. Also, it is very interesting to see that on the right side below, a small amount of cheese is used along with chili. By brushing the region it turns out that it covers *tacos* with *chili* and *cheese* dip.

6 Conclusion

In this paper, we introduced KeywordScape, a visualization tool that implements a novel method for visualizing contextualized keyword embeddings as visual islands. The tool takes advantage of the benefits that contextualized word embeddings bring in contrast to static word embeddings by applying them in a visually searchable contextualized keyword map - a KeywordScape. We implemented a system architecture based on a BERT transformer language model and its ability to represent word meanings. We explained the interaction capabilities that the visualization application provides to the user and illustrated its usability in real-world use cases. Our results show that viewing the meaning of keywords in context leads to new and interesting insights into the document collection, as exemplified by newspapers or cooking recipes.

Acknowledgements

We thank the Michael Stifel Center Jena for funding this work, which is part of the Carl Zeiss Foundation-funded project 'A Virtual Workshop for Digitization in the Sciences' (062017-02).

References

d3-geo-voronoi. https://github.com/Fil/d3-geo-voronoi. Accessed: 2021-06-01.

science-parse. https://github.com/allenai/science-parse. Accessed: 2021-06-01.

spacy. https://github.com/explosion/spaCy. Accessed: 2021-06-01.

umap. https://github.com/lmcinnes/umap. Accessed: 2021-06-01.

M. Berger, Katherine McDonough, and Lee M. Seversky. 2017. cite2vec: Citation-driven document exploration via word embeddings. *IEEE Transactions on Visualization and Computer Graphics*, 23:691–700.

D. Blei, A. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

M. Bostock, Vadim Ogievetsky, and J. Heer. 2011. D³ data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17:2301–2309.

Juntian Chen, Yubo Tao, and Hai Lin. 2018. Visual exploration and comparison of word embeddings. *J. Vis. Lang. Comput.*, 48:178–186.

X. Dai and Robert Prout. 2016. Unlocking super bowl insights: Weighted word embeddings for twitter sentiment classification. *Proceedings of the The 3rd Multidisciplinary International Social Networks Conference on SocialInformatics 2016, Data Science 2016*.

T. Dang and V. Nguyen. 2018. Commodeler: Topic modeling using community detection. In *EuroVA@EuroVis*.

S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the Association for Information Science and Technology*, 41:391–407.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Amanda Gonçalves Dias, E. Milios, and M. C. F. Oliveira. 2019. Trivir: A visualization system to support document retrieval with high recall. *Proceedings of the ACM Symposium on Document Engineering 2019*.

Wenwen Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky. 2013. Hierarchicaltopics: Visually exploring large text collections using topic hierarchies. *IEEE Transactions on Visualization and Computer Graphics*, 19:2002–2011.

Mennatallah El-Assady, Rebecca Kehlbeck, C. Collins, D. Keim, and O. Deussen. 2020. Semantic concept spaces: Guided topic model refinement using word-embedding projections. *IEEE Transactions on Visualization and Computer Graphics*, 26:1001–1011.

Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 1*, 2:559–572.

Ashwinkumar Ganesan, Kianté Brantley, Shimei Pan, and Jian Jhen Chen. 2015. Ldaexplore: Visualizing topic models generated using latent dirichlet allocation. *ArXiv*, abs/1507.06593.

Q. Han, M. John, S. Koch, Ivan Assenov, and T. Ertl. 2018. Labeltransfer - integrating static and dynamic label representation for focus+context text exploration. *2018 International Symposium on Big Data Visual and Immersive Analytics (BDVA)*, pages 1–8.

Florian Heimerl and M. Gleicher. 2018. Interactive analysis of word vector embeddings. *Computer Graphics Forum*, 37.

Florian Heimerl, M. John, Q. Han, S. Koch, and T. Ertl. 2016. Docucompass: Effective exploration of document landscapes. *2016 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 11–20.

Xiaonan Ji, R. Machiraju, Alan Ritter, and P. Yen. 2017. Visualizing article similarities via sparsified article network and map projection for systematic reviews. *Studies in health technology and informatics*, 245:422–426.

Xiaonan Ji, H. Shen, Alan Ritter, R. Machiraju, and P. Yen. 2019. Visual exploration of neural document embedding in information retrieval: Semantics and feature selection. *IEEE Transactions on Visualization and Computer Graphics*, 25:2181–2192.

M. John, Florian Heimerl, Ba-Anh Vu, and T. Ertl. 2018. Visual analysis and exploration of entity relations in document collections. In *VISIGRAPP*.

M. John, K. Kurzhals, and T. Ertl. 2019. Visual exploration of topics in multimedia news corpora. *2019 23rd International Conference Information Visualisation (IV)*, pages 241–248.

T. Kanungo, D. M. Mount, N. Netanyahu, C. D. Piatko, R. Silverman, and A. Wu. 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:881–892.

Minjeong Kim, Kyeongpil Kang, Deok Gun Park, J. Choo, and N. Elmqvist. 2017. Topiclens: Efficient multi-level visual topic exploration of large-scale document collections. *IEEE Transactions on Visualization and Computer Graphics*, 23:151–160.

S. Koch, M. John, M. Wörner, A. Müller, and T. Ertl. 2014. Varifocalreader — in-depth visual analysis of large text documents. *IEEE Transactions on Visualization and Computer Graphics*, 20:1723–1732.

Peter Kraker, Christopher Kittel, and Asura Enkhbayar. 2016. Open knowledge maps: Creating a visual interface to the world's scientific knowledge based on natural language processing.

K. Kucher, R. Martins, C. Paradis, and A. Kerren. 2020. Stancevis prime: visual analysis of sentiment and stance in social media texts. *J. Vis.*, 23:1015–1034.

K. Kucher, Rafael M. Martins, and A. Kerren. 2018a. Analysis of vinci 2009-2017 proceedings. *Proceedings of the 11th International Symposium on Visual Information Communication and Interaction*.

K. Kucher, C. Paradis, and A. Kerren. 2018b. Visual analysis of sentiment and stance in social media texts. In *EuroVis*.

Tuan Manh Le and Leman Akoglu. 2019. Contravis: Contrastive and visual topic modeling for comparing document collections. *The World Wide Web Conference*.

Jiwei Li, Xinlei Chen, Eduard H. Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in nlp. In *HLT-NAACL*.

Q. Li, Kristanto Sean Njotoprawiro, H. Haleem, Qiaoan Chen, Chris Yi, and Xiaojuan Ma. 2018. Embeddingvis: A visual analytics approach to comparative network embedding inspection. *2018 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 48–59.

Shusen Liu, Peer-Timo Bremer, Jayaraman J. Thiagarajan, Vivek Srikumar, Bei Wang, Yarden Livnat, and Valerio Pascucci. 2018. Visual exploration of semantic relationships in neural word embeddings. *IEEE Transactions on Visualization and Computer Graphics*, 24:553–562.

L. V. D. Maaten and Geoffrey E. Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Bodhisattwa Prasad Majumder, Shuyang Li, Jianmo Ni, and Julian McAuley. 2019. Generating personalized recipes from historical user preferences. *arXiv preprint arXiv:1909.00105*.

R. M. Martins, Vasiliki Simaki, K. Kucher, C. Paradis, and A. Kerren. 2017. Stancexplore: Visualization for the interactive exploration of stance in social media.

L. McInnes, J. Healy, and J. Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*.

L. McInnes, John Healy, and S. Astels. 2017. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2:205.

R. Mihalcea and P. Tarau. 2004. Textrank: Bringing order into text. In *EMNLP*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Bhaskar Mitra and Nick Craswell. 2017. Neural text embeddings for information retrieval. *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*.

Piero Molino, Yang Wang, and Jiawei Zhang. 2019. Parallax: Visualizing and understanding the semantics of embedding spaces via algebraic formulae. In *ACL*.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.

Amy L. Olex, Evan French, Peter Burdette, Srilakshmi Sagiraju, Thomas Neumann, Tamas S. Gal, and Bridget T. McInnes. 2021. Topex: Topic exploration of covid-19 corpora.

Deok Gun Park, Seungyeon Kim, Jurim Lee, Jaegul Choo, Nicholas A. Diakopoulos, and Niklas Elmqvist. 2018. Conceptvector: Text visual analytics via interactive lexicon building using word embedding. *IEEE Transactions on Visualization and Computer Graphics*, 24:361–370.

Jeffrey Pennington, R. Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*.

A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

J. Ramos. 2003. Using tf-idf to determine word relevance in document queries.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP/IJCNLP*.

S. Rose, D. Engel, Nick Cramer, and W. Cowley. 2010. Automatic keyword extraction from individual documents.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational linguistics*, 24(1):97–123.

Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, F. Viégas, and M. Wattenberg. 2016. Embedding projector: Interactive visualization and interpretation of embeddings. *ArXiv*, abs/1611.05469.

C. Wang, John W. Paisley, and D. Blei. 2011. Online variational inference for the hierarchical dirichlet process. In *AISTATS*.

X. Wang, Shixia Liu, Junlin Liu, Jianfei Chen, J. Zhu, and B. Guo. 2014. Topicpanorama: A full picture of relevant topics. *IEEE Transactions on Visualization and Computer Graphics*, 22:2508–2521.

Yi Yang, Quanming Yao, and H. Qu. 2017. Vistopic: A visual analytics system for making sense of large document collections using hierarchical topic modeling. *Vis. Informatics*, 1:40–47.

Jiawei Zhang, C. Surakitbanharn, N. Elmqvist, R. Maciejewski, C. Qian, and D. Ebert. 2018. Topotext: Context-preserving text data exploration across multiple spatial scales. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*.

Xinyi Zhou, Apurva Mulay, Emilio Ferrara, and Reza Zafarani. 2020. Recovery: A multimodal repository for covid-19 news credibility research. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*.

A User Evaluation

We evaluated the tool using an active **user study** on the KeywordScape system and collected qualitative and quantitative feedback. The quantitative evaluation was intended to determine the usability and user acceptance of the system. From the qualitative feedback, the top **five** criticisms were extracted and considered as feature requests in the next iteration of the system improvement.

A.1 User Study

The user study was conducted with 24 individuals with scientific backgrounds (54.2% male, 45.8% female). The basic idea was to find out if and how the *KeywordScape* tool supports scientists in their daily research work, which includes examining numerous documents. 45.5% of participants were doctoral students, 29.2% were master's students, 8.3% were postdocs, 12.5% were college professors, and 4.2% were business professionals with an academic background. 29.2% of respondents were under 25 years old, 20.9% were between 25 and 30, and 49.9% were over 30. Study participants' areas of expertise were in Visualization (59.1%), Natural Language Processing (22.7%), Computer Vision, Machine Learning, Augmented Reality (4.5% each), and 4.5% of individuals with other areas of expertise. 29.2% had less than three years of experience in their field, 50% between three and six years, and 20.8% more than six years.

A.1.1 Study Setup

The user study was divided into two parts. To familiarize users with the tool and its use, **six interactive tasks T** had to be solved and participants were asked to rate how difficult on a five-point Likert scale it was to solve each task using the tool. In the second part, users were presented with a **questionnaire** with pictures and statements showing scenes from using the tool. Users were asked to rate the extent to which they agreed with the statements on a five-point Likert scale, based on their previous experience of actively using the tool.

A.1.2 Results

The results of the first part showed positive interaction experiences with the tool, as shown in Figure 7.

T1 was to assess how easy it was to find the grid cell with the highest number of documents in a

	I agree very much	I agree	neutral	I do not agree	I do not agree at all
Overview Creation					
H1: A grid map of document embeddings and their respective keywords can help to get a first overview over the documents of a corpus.	29.2%	66.7%	4.2%	0.0%	0.0%
H2: A visual map of the keywords of a document corpus makes it easy to spot the most relevant topics of a corpus visually by finding the topic islands on the map.	33.3%	25.0%	33.3%	8.3%	0.0%
H3: A representation of a corpus of documents as a visual map makes it easier to navigate a corpus, because islands aggregate semantically similar keywords which then lead to the relevant documents much more convenient than a traditional keyword search.	33.3%	58.3%	8.3%	0.0%	0.0%
Filtering and Relevance Rating					
H4: The islands on a visual map work as visual clues to mark keyword clusters. In combination with brushing this helps to filter out personally relevant documents of a corpus visually by brushing topic regions of interest	45.8%	50.0%	4.2%	0.0%	0.0%
H5: Interactive keyword search using the brushing tool finds more relevant documents than traditional keyword search because it shows me only the documents that contain the keyword of interest in the specific context.	33.3%	37.5%	29.2%	0.0%	0.0%
H6: Interactive keyword search using the brushing tool allows me to find related work for papers covering a specific topic.	29.2%	62.5%	4.2%	4.2%	0.0%
H7: Finding out which documents have the highest amount of their keywords in contexts I am interested in allows me to rate their relevance for my personal research interests.	33.3%	45.8%	16.7%	4.2%	0.0%
Categorization					
H8: A grid map of document embeddings and keywords helps in finding a categorization for the documents of the corpus.	29.2%	54.2%	16.7%	0.0%	0.0%
H9: Filtering documents by brushing keywords of a certain region makes it easier to find a categorization for documents that appear in the same regions. In addition, the tagging feature allows to try out different categorizations.	29.2%	58.3%	12.5%	0.0%	0.0%
Comparison					
H10: Visualizing how a documents' keywords are distributed over all topics of the corpus makes it easier to mentally arrange/localize the content of a document in the corpus.	29.2%	45.8%	25.0%	0.0%	0.0%
H11: Visualizing where a documents keywords are distributed over the semantic map allows to quickly compare the content of a document visually to other documents and find similarities and differences.	25.0%	45.8%	25.0%	4.2%	0.0%

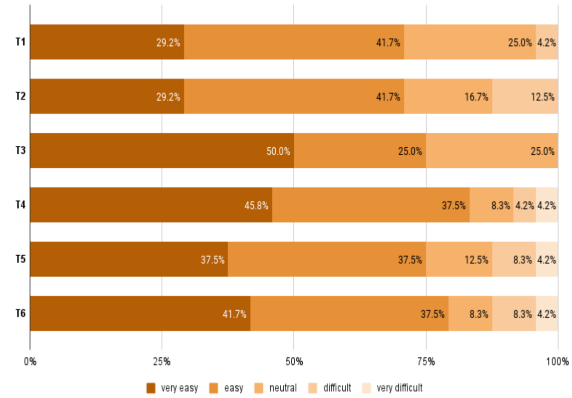

Figure 7: *Top:* The extent to which users agree or disagree with a set of hypotheses **after** completing a series of tasks. *Bottom:* How easy/difficult users consider it to be to perform a particular task.

document map. 70.9% of respondents indicated that this was easy or very easy to achieve with the tool, confirming that the overview map is easy to navigate.

T2 consisted of opening the *contextualized keyword map* and finding a keyword island of personal interest and then zooming into that island to find out more about the specific keyword context. 70.9% of respondents agreed that this was easy to do, 16.7% answered neutral, and 12.5% answered difficult.

T3 was to use the brush tool in the map to filter out the works that cover a particular region and rank those works by personal interest. 75% answered easy or very easy and 25% neutral.

T4 was creating tag groups and adding documents to a corresponding group, which 83.3% rated as easy or very easy.

T5 and T6 was to create a union and an intersection of the created tag groups, which between 75% (union) and 79.2% (intersection) of participants found easy or very easy to achieve.

In the **second part**, users were asked to assess their agreement with **11 hypotheses** stated in relation to the tool's ability to support document *overview generation, relevance estimation, categorization*, and visual document *comparison*. The agreement scores indicate the extent to which the tool can support users in solving these visualization tasks using the interaction capabilities discussed in section 4. The following summarizes user feedback on each exploration task. The results can be seen in Figure 7.

Overview Creation. The majority (95.9%) of respondents agreed that the tool helps to get an overview of relevant topics and that a visual map makes it easier to navigate a large corpus of documents.
Relevance Estimation. Over 90% of participants agreed that the brushing feature makes it easier to find personally relevant documents by outlining contextual areas in a visual map. 70.8% agreed that contextualized keyword search with the brushing tool finds more relevant documents than traditional keyword search, which uses only the number of keywords (without any context).
Categorization. 83.4% agreed that a *document map* helps to find a categorization for the documents in a document collection. 87.5% agreed that brushing regions of a *keyword map* in combination with tagging helps to find a meaningful categorization for the documents in a collection.
Comparison. In terms of document comparability, 70.8% indicated that visualizing the distribution of keywords across the contexts of a document collection allows for quick visual comparison of documents to each other. In addition, 75.0% of respondents agreed that visualizing the distribution of keywords in a document across the semantic landscape facilitates mental retrieval of an individual document within the corpus.

Qualitative Feedback. After assessing the hypotheses, participants were asked to provide qualitative feedback on their main criticisms of the system. From this, the five most frequent points of criticism were extracted:

- performance optimization
- additional keyword projection on grid map
- color legend integration
- additional brushing functionality for papers in the grid map
- BibTeX export

Based on this qualitative feedback, we created feature requests and implemented the desired improvements in response to feedback from the user evaluation. All of the above ideas can now be achieved with the KeywordScape system.