

# WIZMAP: Scalable Interactive Visualization for Exploring Large Machine Learning Embeddings

Zijie J. Wang  
Georgia Tech  
jayw@gatech.edu

Fred Hohman  
Apple  
fredhohman@apple.com

Duen Horng Chau  
Georgia Tech  
polo@gatech.edu

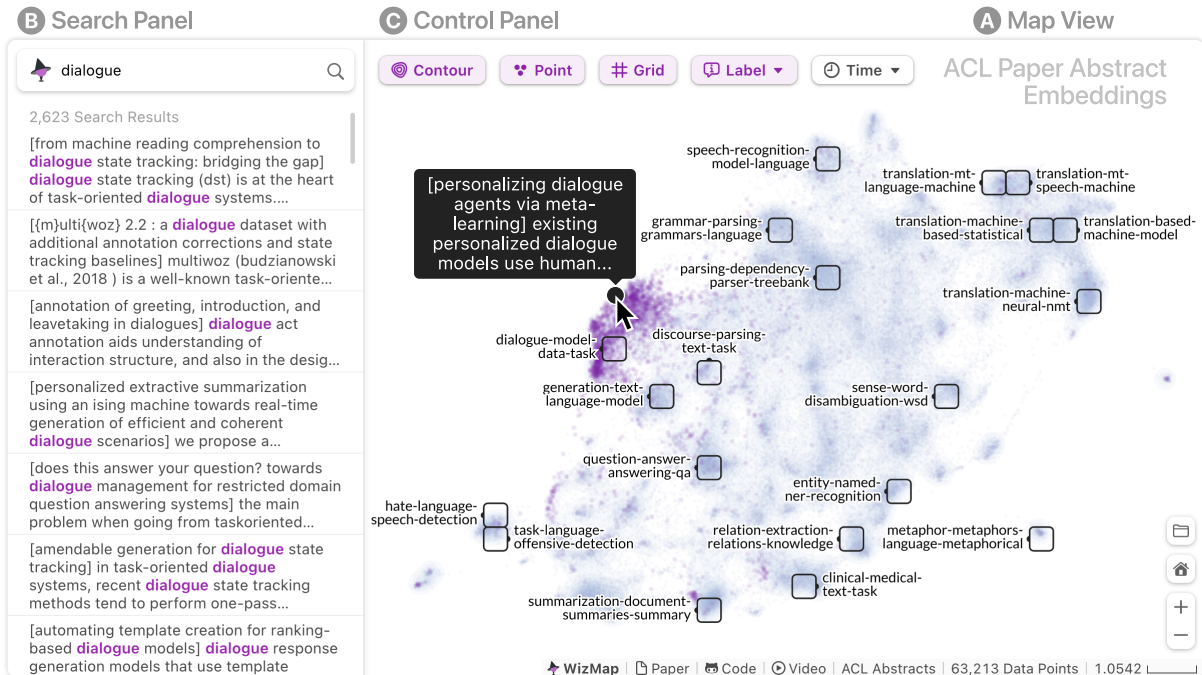


Fig. 1: WIZMAP empowers machine learning researchers and domain experts to easily explore and interpret millions of embedding vectors across different levels of granularity. Consider the task of investigating the embeddings of all 63k natural language processing paper abstracts indexed in ACL Anthology from 1980 to 2022. (A) The Map View tightly integrates a contour layer, a scatter plot, and automatically-generated multi-resolution embedding summaries to help users navigate through the large embedding space. (B) The Search Panel enables users to rapidly test their hypotheses through fast full-text embedding search. (C) The Control Panel allows users to customize embedding visualizations, compare multiple embedding groups, and observe how embeddings evolve over time.

## Abstract

Machine learning models often learn latent embedding representations that capture the domain semantics of their training data. These embedding representations are valuable for interpreting trained models, building new models, and analyzing new datasets. However, interpreting and using embeddings can be challenging due to their opaqueness, high dimensionality, and the large size of modern datasets. To tackle these challenges, we present WIZMAP, an interactive visualization tool to help researchers and practitioners easily explore large embeddings. With a novel multi-resolution embedding summarization method and a familiar map-like interaction design, WIZMAP enables users to navigate and interpret embedding spaces with ease. Leveraging modern web technolo-

gies such as WebGL and Web Workers, WIZMAP scales to millions of embedding points directly in users' web browsers and computational notebooks without the need for dedicated backend servers. WIZMAP is open-source and available at the following public demo link: <https://poloclub.github.io/wizmap>.

## 1 Introduction

Modern machine learning (ML) models learn high-dimensional embedding representations to capture the domain semantics and relationships in the training data (Raghu et al., 2019). ML researchers and domain experts are increasingly using expressive embedding representations to interpret trained models (Park et al., 2022), develop models for new domains (Lee et al., 2018) and modalities (Ben-

younes et al., 2019), as well as analyze and synthesize new datasets (Kern et al., 2016). However, it can be difficult to interpret and use embeddings in practice, as these high-dimensional representations are often opaque, complex, and can contain unpredictable structures (Bolukbasi et al., 2016). Furthermore, analysts face scalability challenges as large datasets can require them to study millions of embeddings holistically (Tang et al., 2016).

To tackle these challenges, researchers have proposed several interactive visualization tools to help users explore embedding spaces (e.g., Smilkov et al., 2016; Liu et al., 2019). These tools often visualize embeddings in a low-dimensional scatter plot where users can browse, filter, and compare embedding points. However, for large datasets, it is taxing or even implausible to inspect embedded data point by point to make sense of the *global structure* of an embedding space. Alternatively, recent research explores using contour plots to summarize embeddings (Sevastjanova et al., 2022; Robertson et al., 2023). Although contour abstractions enable users to obtain an overview of the embedding space and compare multiple embeddings through superposition, a user study reveals that contour plots restrict users’ exploration of an embedding’s *local structures*, where users would prefer to have more visual context (Robertson et al., 2023). To bridge this critical gap between two visualization approaches and provide users with a holistic view, we design and develop WIZMAP (Fig. 1). Our work makes the following **major contributions**:

- **WIZMAP, a scalable interactive visualization tool** that empowers ML researchers and domain experts to explore and interpret embeddings with *millions* of points. Our tool employs a familiar map-like interaction design and fluidly presents adaptive visual summaries of embeddings across different levels of granularity (Fig. 2, § 4).
- **Novel and efficient method to generate multi-resolution embedding summaries.** To automatically summarize embedding neighborhoods with different degrees of granularity, we construct a quadtree (Finkel and Bentley, 1974) from embedding points and extract keywords (text data) or exemplar points (other data types) from tree nodes with efficient branch aggregation (§ 3).
- **An open-source<sup>1</sup> and web-based implementation** that lowers the barrier to interpreting

<sup>1</sup>WIZMAP code: <https://github.com/poloclub/wizmap>

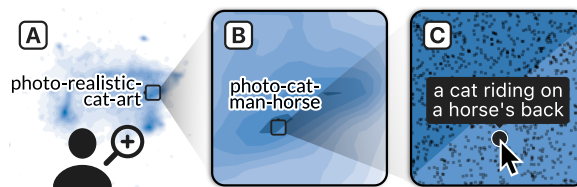


Fig. 2: WIZMAP enables users to explore embeddings at different levels of detail. (A) The contour plot with automatically-generated embedding summaries provides an overview. (B) Embedding summaries adjust in resolution as users zoom in. (C) The scatter plot enables the investigation of individual embeddings.

and using embeddings. We develop WIZMAP with modern web technologies such as WebGL and Web Workers so that anyone can access the tool directly in both their web browsers and computational notebooks without a need for dedicated backend servers (§ 4.4). For a demo video of WIZMAP, visit <https://youtu.be/8fJG87QVceQ>.

## 2 Background and Related Work

Researchers can extract a data point’s embeddings by collecting its corresponding layer activations in neural networks trained for specific tasks such as classification and generation (Raghu et al., 2019). Additionally, researchers have developed task-agnostic models, such as word2vec (Mikolov et al., 2013), ELMo (Peters et al., 2018), and CLIP (Radford et al., 2021) that generate transferable embeddings directly. These embeddings have been shown to outperform task-specific, state-of-the-art models in downstream tasks (Radford et al., 2021; Dwivedi et al., 2021).

### 2.1 Dimensionality Reduction

Embeddings are often high-dimensional, such as 300-dimensions for word2vec, or 768-dimensions for CLIP and BERT Base (Devlin et al., 2018). Therefore, to make these embeddings easier to visualize, researchers often apply dimensionality reduction techniques to project them into 2D or 3D space. Some popular dimensionality reduction techniques include UMAP (McInnes et al., 2020), t-SNE (van der Maaten and Hinton, 2008), and PCA (Pearson, 1901). Each of these techniques has its own strengths and weaknesses in terms of how well it preserves the embeddings’ global structure, its stochasticity, interpretability, and scalability. Despite these differences, all dimensionality reduction techniques produce data in the same structure. This

means users can choose any technique and visualize the projected embeddings with WIZMAP.

## 2.2 Interactive Embedding Visualization

Researchers have introduced interactive visualization tools to help users explore embeddings (e.g., Liu et al., 2018; Li et al., 2018; Arendt et al., 2020). For example, Embedding Projector (Smilkov et al., 2016) allows users to zoom, rotate, and pan 2D or 3D projected embeddings to explore and inspect data point features. Similarly, Deepscatter (Schmidt, 2021) and regl-scatterplot (Lekschas, 2023) empowers users to explore billion-scale 2D embeddings in their browsers. Latent Space Cartography (Liu et al., 2019) helps users find and refine meaningful semantic dimensions within the embedding space. In addition, researchers have designed visualizations to aid users in comparing embeddings, such as embComp (Heimerl et al., 2022) visualizing local and global similarities between two embeddings, Emblaze (Sivaraman et al., 2022) tracing the changes in the position of data points across two embeddings, and Embedding Comparator (Boggust et al., 2022) highlighting the neighborhoods around points that change the most across embeddings. In contrast, WIZMAP aims to help users navigate and interpret both the global and local structures of large embedding spaces by offering visual contexts at varying levels of granularity.

## 3 Multi-scale Embedding Summarization

Researchers have highlighted users’ desire for embedding visualizations to provide visual contexts and embedding summaries to facilitate exploration of various regions within the embedding space (Robertson et al., 2023). However, generating embedding summaries is challenging for two reasons. First, efficiently summarizing millions of data points in larger datasets can be a formidable task. Second, selecting the embedding regions to summarize is difficult, as users possess varying interests in regions of different sizes and levels of granularity. To tackle this challenge, we propose a novel method to automatically generate multi-resolution embedding summaries at scale.

**Multi-resolution Quadtree Aggregation.** First, we apply a dimensionality reduction technique such as UMAP to project high-dimensional embedding vectors into 2D points. From these points, we construct a quadtree (Finkel and Bentley, 1974), a tree data structure that recursively partitions a 2D space

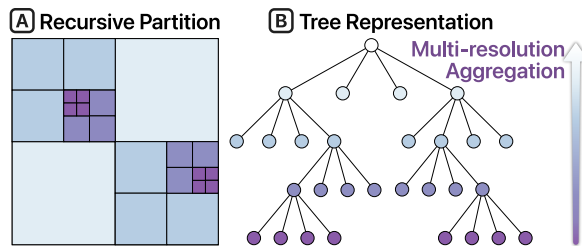


Fig. 3: (A) A quadtree recursively partitions a 2D space into four equally-sized squares, (B) and each square is represented as a tree node. WIZMAP efficiently aggregates information from the leaves to the root, summarizing embeddings at different levels of granularity.

into four equally-sized squares, each represented as a node. Each data point exists in a unique leaf node. To summarize embeddings across different levels of granularity, we traverse the tree bottom up. In each iteration, we first extract summaries of embeddings in each leaf node, and then merge the leaf nodes at the lowest level with their parent node. This process continues recursively, with larger and larger leaf nodes being formed until the entire tree is merged into a single node at the root. Finally, we map pre-computed embedding summaries to a suitable granularity level and dynamically show them as users zoom in or out in WIZMAP (§ 4.1).

**Scalable Leaf-level Summarization.** When performing quadtree aggregation, researchers have the flexibility to choose any suitable method for summarizing embedding from leaf nodes. For text embeddings, we propose t-TF-IDF (tile-based TF-IDF) that adapts TF-IDF (term frequency-inverse document frequency) to extract keywords from leaf nodes (Sparck Jones, 1972). Our approach is similar to c-TF-IDF (classed-based TF-IDF) that combines documents in a cluster into a meta-document before computing TF-IDF scores (Grootendorst, 2022). Here, we merge all documents in each leaf node (i.e., a tile in the quadtree partition) as a meta-document and compute TF-IDF scores across all leaf nodes. Finally, we extract keywords with the highest t-TF-IDF scores to summarize embeddings in a leaf node. This approach is scalable and complementary to quadtree aggregation. Because our document merging is hierarchical, we only construct the n-gram count matrix once and update it in each aggregation iteration with just one matrix multiplication. Summarizing 1.8 million text embeddings across three granularity levels takes only about 55 seconds on a MacBook Pro. For non-text data, we summarize embeddings by finding points closest to the embedding centroid in a leaf node.



## 4 User Interface

Leveraging pre-computed multi-resolution embedding summarization (§ 3), WIZMAP tightly integrates three interface components (Fig. 1A–C).

### 4.1 Map View

The *Map View* (Fig. 1A) is the primary view of WIZMAP. It provides a familiar map-like interface that allows users to pan and zoom to explore different embedding regions with varying sizes. To help users easily investigate both the global structure and local neighborhoods of their embeddings, the *Map View* integrates three layers of visualization.

**Distribution Contour.** To provide users with a quick overview of the global structure of their embeddings, we use Kernel Density Estimation (KDE) (Rosenblatt, 1956) to estimate the distribution of 2D embedding points. We use a standard multivariate Gaussian kernel with a Silverman bandwidth for the KDE model (Silverman, 2018). Next, we compute the distribution likelihoods over a  $200 \times 200$  2D grid whose size is determined by the range of all embedding points. Finally, we visualize the likelihoods over the grid as a contour plot (Fig. 4), highlighting the high-level density distribution of users’ embeddings. Researchers can adjust the grid density, and we tune it by balancing the computation time and the contour resolution.

**Multi-resolution Labels.** The *Map View* helps users interpret embeddings across various levels of granularity by dynamically providing pre-computed contextual labels. It overlays summaries generated via quadtree aggregation (§ 3) onto the distribution contour and scatter plot. Users can hover over to see the summary from a quadtree tile closest to the cursor. Our tool adjusts the label’s tile size based on the user’s current zoom level. For example, when a user zooms into a small region, the *Map View* shows summaries computed at a lower level in the quadtree. In addition to on-demand embedding summaries, this view also automatically labels high-density regions (Fig. 4) by showing summaries from quadtree tiles near the geometric centers of high-probability contour polygons.

**Scatter Plot.** To help users pinpoint embeddings within their local neighborhoods, the *Map View* visualizes all embedding points in a scatter plot with their 2D positions.

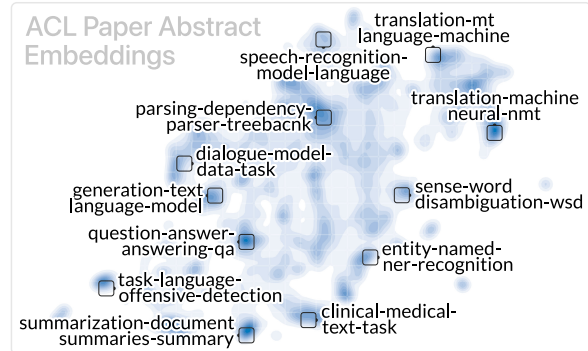
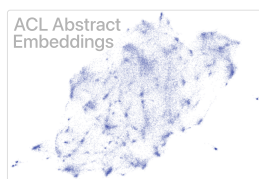
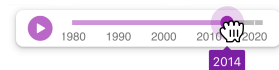


Fig. 4: The *Map View* provides an embedding overview via a contour plot and auto-generated multi-resolution embedding labels placed around high-density areas.

The corresponding scatter plot for Fig. 4 is shown on the right. Users can specify the color of each embedding point to encode additional features, such as the class of embeddings. Also, users can hover over an embedding point to reveal its original data, such as ACL paper abstracts (§ 5.1).

### 4.2 Control Panel

The *Map View* shows all three visualization layers



by default, and users can customize them to fit their needs by clicking buttons in the *Control Panel* (Fig. 1C). In addition, WIZMAP allows users to compare multiple embedding groups in the same embedding space by superimposing them in the *Map View* (Gleicher, 2018). In the case of embeddings that include times, users can use a slider (shown on the right) in the *Control Panel* to observe changes in the embeddings over time (Fig. 5).

### 4.3 Search Panel

Searching and filtering can help users discover interesting embedding patterns and test their hypothesis regarding the embedding structure (Carter et al., 2019). In WIZMAP, users can use the *Search Panel* (Fig. 1B) to search text embeddings including specified words in the original data. The panel shows a list of search results, and the *Map View* highlights their corresponding embedding points.

### 4.4 Scalable & Open-source Implementation

WIZMAP is scalable to *millions* of embedding points, providing a seamless user experience with zooming and animations, all within web browsers without backend servers. To achieve this, we leverage modern web technologies, especially WebGL to render embedding points with the `regl` API (Ly-senko, 2016). We also use `Web Workers` and `Streams API` to enable the streaming of large em-

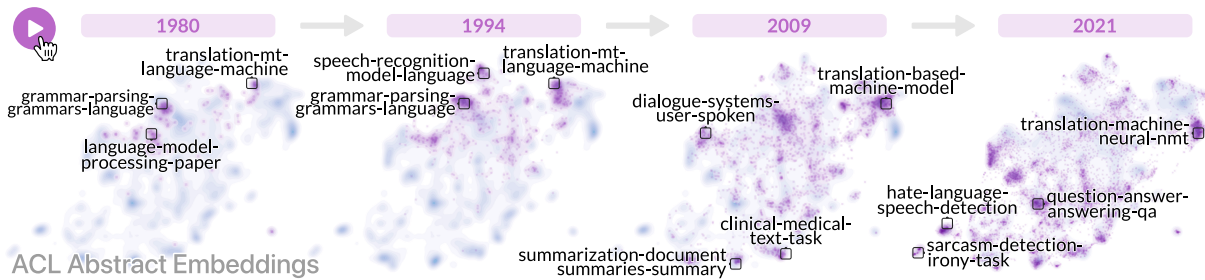



Fig. 5: WIZMAP allows users to observe how embeddings change over time. For example, when exploring 63k ACL paper abstracts, clicking the play button  in the *Control Panel* animates the visualizations to show embeddings of papers published in each year in purple and the distribution of all papers in blue. This animation highlights changes in ACL research topics over time, such as the decline in popularity of grammar and the rise of question-answering.

bedding files in parallel with rendering. To enable fast full-time search, we apply a contextual indexing algorithm with FlexSearch (Wilkerling, 2019). We use D3 (Bostock et al., 2011) for other visualizations and scikit-learn (Pedregosa et al., 2011) for KDE. To ensure that our tool can be easily incorporated into users’ current workflows, we apply NOVA (Wang et al., 2022b) to make WIZMAP available within computational notebooks. We provide detailed tutorials to help users use our tool with their embeddings. We have open-sourced our implementation to support future research and development of embedding exploration tools.

## 5 Usage Scenarios

We present two hypothetical scenarios, each with real embedding data, to demonstrate how WIZMAP can help ML researchers and domain experts easily explore embeddings and gain a better understanding of ML model behaviors and dataset patterns.


### 5.1 Exploring ACL Research Topic Trends

Helen, a science historian, is interested in exploring the evolution of computational linguistic and natural language processing (NLP) research since its inception. She downloads the Bibtex files of all papers indexed in ACL Anthology (Rohatgi, 2022), and extracts the paper title and abstract from 63k papers that have abstracts available. Then, Helen applies MPNet, a state-of-the-art embedding model (Song et al., 2020), to transform the concatenation of each paper’s title and abstract into a 768-dimensional embedding vector. She then trains a UMAP model to project extracted embeddings into a 2D space. She tunes the UMAP’s hyperparameter  $n\_neighbors$  to ensure projected points are spread out (Coenen and Pearce, 2019).

Helen uses a Python function provided by WIZMAP to generate three JSON files containing embedding summaries (§ 3), the KDE distribu-

tions (§ 4.1), and the original data in a streamable format (Hoeger et al., 2014). Helen configures the function to use the dataset’s year feature as the embedding’s time—the function computes the KDE distribution of embeddings for each year slice. She provides the files to WIZMAP and sees a visualization of all ACL abstract embeddings (Fig. 4A).

**Embedding Exploration.** In the *Map View*, Helen explores embeddings with zoom and pan. She also uses the *Search Panel* to find papers with specific keywords, such as “dialogue”, and Helen is pleased to see all related papers are grouped in a cluster (Fig. 1B). With the help of multi-resolution embedding summaries, Helen quickly gains an understanding of the structure of her embedding space. For example, she finds that the top right cluster features translation papers while the lower clusters feature summarization and medical NLP.

**Embedding Evolution** To examine how ACL research topics change over time, Helen clicks the play button clicking the play button  in the *Control Panel* to animate the visualizations. The *Map View* shows embeddings of papers published in each year from 1980 to 2022 in purple, while the distribution of all papers is shown as a blue background (Fig. 5). As Helen observes the animation, she identifies several interesting trends. For example, she observes a decline in the popularity of grammar research, while question-answering has become increasingly popular. She also notes the emergence of some small clusters in recent years, featuring relatively new topics, such as sarcasm, humor, and hate speech. Satisfied with the findings using WIZMAP, Helen decides to write an essay on the trend of NLP research over four decades.

### 5.2 Investigating Text-to-Image Model Usage

Bob, an ML researcher, works on improving text-to-image generative models. Recent advancements

in diffusion models, such as Stable Diffusion (Rom-bach et al., 2022), have attracted an increasing number of users to generate photorealistic images by writing text prompts. To gain an understanding of these models’ behaviors and identify potential weaknesses for improvement, Bob decides to study how users use these models in the wild by analyzing DiffusionDB, a dataset containing 14 million images generated by Stable Diffusion with 1.8 million unique text prompts (Wang et al., 2022a).

Bob’s analysis goal is to study the relationship between the text prompts and their generated images. Thus, he chooses to use CLIP (Radford et al., 2021) to encode both prompts and images into a 768-dimensional multimodal embedding before projecting them to a 2D space with UMAP. He uses prompts to generate embedding summaries for the CLIP space. After creating all JSON files, WIZMAP visualizes all 3.6 million embeddings (Fig. 6).

**Embedding Exploration.** Bob begins his exploration by hiding image embeddings and scatter plots, focusing on the global structure of embeddings with the contour plot and embedding summaries. He discovers two dominant prompt categories: art-related prompts and photography-related prompts. Two categories appear far from each other, which is not surprising as they are expected to have distinct semantic representations. Bob also notices two smaller clusters within the photography region, prompting him to zoom in and turn on the scatter plot to further investigate these local regions (Fig. 2). After hovering over a few points, he realizes one cluster is mostly about non-human objects while the other is about celebrities.

**Embedding Comparison.** To investigate the relationship between text prompts and their generated images, Bob clicks a button in the *Control Panel* to superimpose the contour and scatter plot of image embeddings in red onto the text embedding visualizations in blue (Fig. 6). Bob quickly identifies areas where two distributions overlap and differ. He notes that the “movie” cluster in the text embeddings has a lower density in the image embeddings, whereas a high-density “art portrait” cluster emerges in image embeddings. Bob hypothesizes that Stable Diffusion may have limited capability to generate photorealistic human faces (Borji, 2022). After exploring embedding with WIZMAP, Bob is pleased with his findings, and he will apply his insights to improve the curation of his training data.

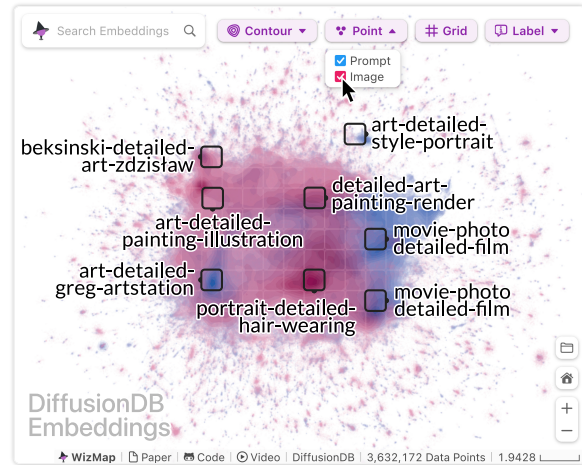


Fig. 6: WIZMAP enables users to compare multiple embeddings by visualization superposition. For instance, comparing the CLIP embeddings of **1.8 million** Stable Diffusion **prompts** and **1.8 million** generated **images** reveals key differences between two distributions.

## 6 Future Work and Conclusion

WIZMAP integrates a novel quadtree-based embedding summarization technique that enables users to easily explore and interpret large embeddings across different levels of granularity. Our usage scenarios showcase our tool’s potential for providing ML researchers and domain experts with a holistic view of their embeddings. Reflecting on our design and development of WIZMAP, we acknowledge its limitations and distill future research directions that could further assist users in interpreting and applying embeddings for downstream tasks.

- **User evaluation.** To investigate the usefulness of flexible transitioning across various levels of abstraction during embedding exploration, future researchers can use WIZMAP as a research instrument to conduct observational user studies with ML researchers and domain experts.
- **Automated insights.** Our tool provides automatic and multi-scale visual contexts to guide users in their exploration. While our quadtree-based approach is effective and scalable, it is sensitive to tile size selection. Researchers can explore more robust methods for embedding summarization and automated data insights, such as clustering-based approaches (Law et al., 2020).
- **Enhanced comparison.** WIZMAP helps users compare embedding groups through contour superposition. However, for local comparisons, other techniques such as juxtaposition and explicit encoding may be more effective (Gleicher, 2018). Future researchers can design visualization tools that integrate these techniques.



## 7 Broader Impact

We designed and develop WIZMAP with good intentions—to help ML researchers and domain experts easily explore and interpret large embeddings. However, bad actors could exploit insights gained from using WIZMAP for malevolent purposes. For example, research has shown that ML embeddings contain societal biases (Bolukbasi et al., 2016). Therefore, bad actors could manipulate and sabotage ML predictions by injecting inputs whose embeddings are known to associate with gender and racial biases. The potential harms of biased embeddings warrant further study.

## Acknowledgements

We thank our anonymous reviewers for their insightful comments. This work was supported in part by a J.P. Morgan PhD Fellowship, Apple Scholars in AI/ML PhD fellowship, DARPA GARD, gifts from Cisco, Bosch, and NVIDIA. Use, duplication, or disclosure is subject to the restrictions as stated in Agreement number HR00112030001 between the Government and the Performer.

## References

- Dustin L. Arendt, Nasheen Nur, Zhuanyi Huang, Gabriel Fair, and Wenwen Dou. 2020. [Parallel embeddings: A visualization technique for contrasting learned representations](#). In *Proceedings of the 25th International Conference on Intelligent User Interfaces*.
- Hedi Ben-younes, Remi Cadene, Nicolas Thome, and Matthieu Cord. 2019. [BLOCK: Bilinear Superdiagonal Fusion for Visual Question Answering and Visual Relationship Detection](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 33.
- Angie Boggust, Brandon Carter, and Arvind Satyanarayan. 2022. [Embedding Comparator: Visualizing Differences in Global Structure and Local Neighborhoods via Small Multiples](#). In *27th International Conference on Intelligent User Interfaces*.
- Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. [Man is to computer programmer as woman is to homemaker? Debiasing word embeddings](#). In *Advances in Neural Information Processing Systems*, volume 29.
- Ali Borji. 2022. [Generated Faces in the Wild: Quantitative Comparison of Stable Diffusion, Midjourney and DALL-E 2](#). *arXiv 2210.00586*.
- Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. [D<sup>3</sup> Data-Driven Documents](#). *IEEE TVCG*, 17.
- Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. 2019. [Activation Atlas](#). *Distill*, 4.
- Andy Coenen and Adam Pearce. 2019. [Understanding UMAP](#).
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](#). *arXiv:1810.04805 [cs]*.
- Debidatta Dwivedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. 2021. [With a little help from my friends: Nearest-neighbor contrastive learning of visual representations](#). In *ICCV*.
- R. A. Finkel and J. L. Bentley. 1974. [Quad trees a data structure for retrieval on composite keys](#). *Acta Informatica*, 4.
- Michael Gleicher. 2018. [Considerations for Visualizing Comparison](#). *IEEE Transactions on Visualization and Computer Graphics*, 24.
- Maarten Grootendorst. 2022. [BERTopic: Neural topic modeling with a class-based TF-IDF procedure](#). *arXiv preprint arXiv:2203.05794*.
- Florian Heimerl, Christoph Kralj, Torsten Moller, and Michael Gleicher. 2022. [embComp : Visual Interactive Comparison of Vector Embeddings](#). *IEEE Transactions on Visualization and Computer Graphics*, 28.
- Thorsten Hoeger, Chris Dew, Finn Pauls, and Jim Wilson. 2014. [Newline Delimited JSON: A standard for delimiting JSON in stream protocols](#).
- Margaret L. Kern, Gregory Park, Johannes C. Eichstaedt, H. Andrew Schwartz, Maarten Sap, Laura K. Smith, and Lyle H. Ungar. 2016. [Gaining insights from social media language: Methodologies and challenges](#). *Psychological Methods*, 21.
- Po-Ming Law, Alex Endert, and John Stasko. 2020. [Characterizing Automated Data Insights](#). In *2020 IEEE Visualization Conference (VIS)*.
- Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. 2018. [CleanNet: Transfer learning for scalable image classifier training with label noise](#). In *CVPR*.
- Fritz Lekschas. 2023. [Regl-Scatterplot: A Scalable Interactive JavaScript-based Scatter Plot Library](#). *Journal of Open Source Software*, 8.
- Quan Li, Kristanto Sean Njotoprawiro, Hammad Haleem, Qiaoan Chen, Chris Yi, and Xiaojuan Ma. 2018. [EmbeddingVis: A Visual Analytics Approach to Comparative Network Embedding Inspection](#). *arXiv:1808.09074 [cs]*.
- 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.
- Yang Liu, Eunice Jun, Qisheng Li, and Jeffrey Heer. 2019. [Latent Space Cartography: Visual Analysis of Vector Space Embeddings](#). *Computer Graphics Forum*, 38.
- Mikola Lysenko. 2016. [Regl: Functional WebGL](#).
- Leland McInnes, John Healy, and James Melville. 2020. [UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction](#). *arXiv:1802.03426 [cs, stat]*.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. [Efficient Estimation of Word Representations in Vector Space](#). *arXiv 1301.3781*.
- Haekyu Park, Nilaksh Das, Rahul Duggal, Austin P. Wright, Omar Shaikh, Fred Hohman, and Duen Horng Polo Chau. 2022. [NeuroCartography: Scalable Automatic Visual Summarization of Concepts in Deep Neural Networks](#). *IEEE Transactions on Visualization and Computer Graphics*, 28.
- Karl Pearson. 1901. [On lines and planes of closest fit to systems of points in space](#). *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. [Scikit-learn: Machine learning in python](#). *the Journal of machine Learning research*, 12.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *NAACL HLT*.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. [Learning transferable visual models from natural language supervision](#). In *ICML*, volume 139 of *Proceedings of Machine Learning Research*.
- Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. 2019. [Transfusion: Understanding transfer learning for medical imaging](#). In *Advances in Neural Information Processing Systems*, volume 32.
- Samantha Robertson, Zijie J. Wang, Dominik Moritz, Mary Beth Kery, and Fred Hohman. 2023. [Angler: Helping Machine Translation Practitioners Prioritize Model Improvements](#). In *CHI Conference on Human Factors in Computing Systems*.
- Shaurya Rohatgi. 2022. [ACL anthology corpus with full text](#). Github.
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. [High-resolution image synthesis with latent diffusion models](#). In *CVPR*.
- Murray Rosenblatt. 1956. [Remarks on Some Nonparametric Estimates of a Density Function](#). *The Annals of Mathematical Statistics*, 27.
- Benjamin Schmidt. 2021. [Deepscatter: Zoomable, animated scatterplots in the browser that scales over a billion points](#).
- Rita Sevastjanova, Eren Cakmak, Shauli Ravfogel, Ryan Cotterell, and Mennatallah El-Assady. 2022. [Visual Comparison of Language Model Adaptation](#). *IEEE Transactions on Visualization and Computer Graphics*.
- Bernard W Silverman. 2018. [Density Estimation for Statistics and Data Analysis](#).
- Venkatesh Sivaraman, Yiwei Wu, and Adam Perer. 2022. [Emblaze: Illuminating Machine Learning Representations through Interactive Comparison of Embedding Spaces](#). In *27th International Conference on Intelligent User Interfaces*.
- Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B. Viégas, and Martin Wattenberg. 2016. [Embedding Projector: Interactive Visualization and Interpretation of Embeddings](#). *arXiv 1611.05469*.
- Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tiejun Liu. 2020. [MpNet: Masked and permuted pre-training for language understanding](#). *Advances in Neural Information Processing Systems*, 33.
- Karen Sparck Jones. 1972. [A statistical interpretation of term specificity and its application in retrieval](#). *Journal of Documentation*, 28.
- Jian Tang, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei. 2016. [Visualizing Large-scale and High-dimensional Data](#). In *Proceedings of the 25th International Conference on World Wide Web*.
- Laurens van der Maaten and Geoffrey Hinton. 2008. [Visualizing data using t-SNE](#). *Journal of Machine Learning Research*, 9.
- Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. 2022a. [DiffusionDB: A large-scale prompt gallery dataset for text-to-image generative models](#). *arXiv:2210.14896 [cs]*.
- Zijie J. Wang, David Munechika, Seongmin Lee, and Duen Horng Chau. 2022b. [NOVA: A Practical Method for Creating Notebook-Ready Visual Analytics](#). *arXiv:2205.03963 [cs]*.
- Thomas Wilkerling. 2019. [FlexSearch: Next-Generation full text search library for Browser and Node.js](#).