BeyondFederated: truly decentralised learning at the edge

Quinten van Eijs Delft University of Technology Delft, The Netherlands

I. Introduction

While centralized approaches to data storage and computation dominate the current landscape of Artificial Intelligence (AI) and Machine Learning (ML), the increasing demand for privacy, security, and scalability necessitates a shift towards decentralized learning paradigms. However, existing decentralized learning methods often suffer from fragmented knowledge due to isolated training, static embeddings that fail to adapt to evolving data, and privacy concerns stemming from data sharing requirements.

Therefore, this research aims to investigate how peer-to-peer (P2P) vector databases can be leveraged to address the challenges of decentralized learning by:

1) **Enabling collaborative learning:** Developing robust communication protocols and efficient algorithms for nodes to share data and model updates, fostering the collective refinement of knowledge and leading to better, more consistent models across the network.
2) **Facilitating dynamic embedding learning:** Implementing mechanisms for nodes to dynamically adjust vector representations based on their local data and interactions, resulting in contextually relevant and personalized models that adapt to the evolving data landscape.
3) **Ensuring privacy-preserving learning:** Exploring privacy-preserving techniques like federated learning and homomorphic encryption to allow nodes to contribute to the learning process without directly sharing sensitive data, thereby addressing privacy concerns inherent in decentralized learning.

By tackling these challenges, this research aims to unlock the full potential of P2P vector databases for decentralized learning, paving the way for a more secure, scalable, and privacy-conscious future of AI and ML applications.

II. Problem Description

Highlight the limitations of current content retrieval systems on platforms like Youtube and TikTok. Introduce the concept of P2P vector databases and dynamic embeddings as potential solutions. Formulate the research objectives and outline the thesis structure. **this section definately need more focus** In the era of big data and collaborative learning, the ability to efficiently search and retrieve information from large-scale, high-dimensional datasets is crucial. This is particularly true in the context of peer-to-peer vector databases, where data is distributed across multiple nodes and the search for similar vectors (i.e., nearest neighbors) is a common operation.

In the field of music, collaborative learning systems often involve the sharing and comparison of high-dimensional data vectors, such as musical profiles, song features, or user listening habits. The ability to quickly find the most similar vectors - the nearest neighbors - is key to many features of these systems, such as personalized music recommendations, grouping of similar music genres, or detection of music trends.

Traditional exact nearest neighbor search algorithms, while accurate, suffer from the ”curse of dimensionality” and do not scale well with the size and dimensionality of the dataset. This leads to significant computational overhead and latency in retrieving the nearest neighbors, which is not feasible in real-time applications or systems with large volumes of data.

This is where Approximate Nearest Neighbor (ANN) algorithms come into play. ANN algorithms trade off a small amount of accuracy for a substantial increase in speed and scalability. They provide a practical solution for nearest neighbor search in high-dimensional spaces, enabling faster query times and lower resource usage.

However, the importance and necessity of ANN in peer-to-peer vector databases and collaborative learning systems in music is not universally recognized. There is a need to raise awareness about the benefits of ANN, and to encourage its adoption in the design and implementation of these systems. This will not only improve the efficiency and performance of these systems, but also enable them to handle larger and more complex datasets, thereby unlocking new possibilities in data analysis, machine learning applications, and personalized, collaborative music experiences.

III. Background and Related Works

A. Peer to Peer Data accessibility

TODO: Small section about disruptive way of accessing (personalized) information without requiring central entities which can be targetted by authorities

B. Decentralized Learning

TODO: Federated learning approaches always require a central entity to manage the model. Include resources about personalized model updates which are then distibuted

C. Approximate Nearest Neighbors

TODO: Explain why Approximate Nearest Neighbors should be pointed out is a well studied problem one of the biggest limitations why systems always require central entity due to calculation, following state of the art ANN-appoaches: **Spotify - Annoy** is a C++ library to search for points in space that are close to a given query point. It also creates large read-only file-based data structures that are mmapped into memory. It is built and used by Spotify for music recommendations to find similar music tracks based on their audio features.

Annoy uses random projections and tree structures to implement approximate nearest neighbor search, which allows it to handle high-dimensional data efficiently. It trades off a small amount of accuracy for a substantial increase in speed and scalability, making it suitable for large-scale datasets and real-time applications. https://github.com/spotify/annoy

FAISS (Facebook AI Similarity Search) is a library developed by Facebook's AI Research team for efficient similarity search and clustering of high-dimensional vectors. It's particularly useful for tasks that involve searching through large datasets to find items similar to a given item, which is a common operation in machine learning and data analysis.

FAISS uses a technique called Approximate Nearest Neighbor (ANN) search to quickly find the vectors in a dataset that are most similar to a given query vector. It's designed to handle very large datasets, potentially containing billions of vectors, and it can be significantly faster than traditional exact search methods. https://github.com/facebookresearch/faiss

Scann (Scalable Nearest Neighbors) is a library developed by Google Research for efficient vector similarity search. It's designed to handle large-scale, high-dimensional datasets, making it useful for tasks that involve finding items in a dataset that are most similar to a given item.

Scann uses an approach called Approximate Nearest Neighbors (ANN) search to quickly find the vectors in a dataset that are most similar to a given query vector. It's optimized for both dense (like word embeddings) and sparse datasets (like bag-of-words), and it can be significantly faster and more resource-efficient than traditional exact search methods. https://github.com/google-research/google-research/tree/master/scann

ANN-Benchmarks is a benchmarking environment for approximate nearest neighbor algorithms search, https://ann-benchmarks.com/.

IV. PeerAI

TODO: Refactor PeerAI to a more research suited name..

A. *Describe the system architecture, including data partitioning, communication protocols, and consistency management mechanisms.*

- Describe the system architecture, including data partitioning, communication protocols, and consistency management mechanisms.
- Address scalability, fault tolerance, and security aspects of the P2P network.
- Adress P2P functional requirements using SuperAPP! Highlight choice of using SCaNN + tensorflow mobile.
- Chosen Embedding model and possible variations.
- Discuss the role of individual nodes and their interactions within the network.
- Important to note that current setup has integration with Machine Learning Frameworks.
- Architecture descripton + diagram.

B. *Dynamic Embedding Learning*

- Explain the concept of dynamic embeddings and their benefits for personalized content retrieval.
- Describe the proposed algorithms for collaborative learning within the P2P network.
- Consider user privacy and security during embedding updates and sharing

C. *Privacy Analysis and Security*

- Discuss potential privacy risks associated with P2P content retrieval and embedding learning.
- Evaluate the effectiveness of implemented privacy-preserving techniques.
- Propose further measures to enhance user data privacy and security.

V. Evaluation and Experiments

- Design and conduct experiments comparing the proposed system with existing approaches.
- Evaluate search accuracy, efficiency, and personalization based on real-world datasets.
- Analyze the impact of dynamic embeddings and collaborative learning on search performance.

A. *Vector Space*

Indexing Youtube and TikTok: Explore the feasibility of indexing content from these platforms using P2P vector databases. Consider potential challenges related to copyright, privacy, and data access limitations. Recommending 20k Closely Related Items: Design and evaluate algorithms that recommend highly relevant content based on a user's personalized model and collaborative learning within the P2P network. Analyze the accuracy and diversity of recommendations compared to traditional approaches.

- Pretrained 20k item model including 140 partitions and 128 embedded asymmetric hashes
- Inference Speed
- Impact of different embedding methods and inpact on clusters. Such as sentence encoder vs weighted song features.
- cluster visualization

B. *Investigate the use of various distance metrics and similarity search algorithms for enhanced retrieval accuracy.*

C. *Benchmark against traditional or static vector databases database?*

D. *(CPU/GPU) Performance*

VI. Conclusion and Future Work

- Summarize the key findings and contributions.
- Discuss limitations and potential future research directions.
- Highlight the broader implications of P2P vector databases and dynamic embeddings for personalized content retrieval and other applications.

Current setup has no block/filter on given other nodes such that all data and given meta data is learned by peers even unwanted metadata. This is out of scope for the given research.