UNIVERSITÀ DEGLI STUDI DI PADOVA

DEPARTMENT OF MATHEMATICS "TULLIO LEVI-CIVITA"

MASTER THESIS IN DATA SCIENCE

EMAIL MINING TO UNCOVER AUTOMATION OPPORTUNITIES

SUPERVISOR
PROF. LIVIO FINOS
UNIVERSITÀ DEGLI STUDI DI PADOVA

CO-SUPERVISOR
PROF. LLUÍS PADRÓ CIRERA
UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER CANDIDATE
ANNA PUTINA

STUDENT ID
2081379

ACADEMIC YEAR
2024-2025

Abstract

In the modern working environment, people are overloaded with data. We collect, store, and process vast amounts of information for business, safety, and legal purposes. Managing this information manually is increasingly challenging, and employees spend significant time performing repetitive tasks. Consequently, automation offers a promising breakthrough that could substantially reduce costs.

This project leverages data mining on email data, initially focusing on emails with the potential to extend findings to other areas. We aim to identify patterns within email interactions to automate repetitive tasks such as reading, responding, attaching files, forwarding information, and managing spam. This automation could significantly enhance productivity and user satisfaction by reducing the manual effort involved in email management.

Our approach involves analyzing a large dataset of business emails to detect recurring interaction patterns. To reconstruct email chains and convert them into structured data, we use a systematic methodology relying on the emails' metadata and content. After automatically processing the texts, we generate embeddings to convert the text into numerical representations. A time-aware distance metric assesses sequence similarity to cluster the emails, revealing potential automation opportunities.

The results demonstrate the feasibility of extracting processes and similar interactions from emails using the proposed solution. This serves as a model pipeline for future projects, where specific steps can be adapted to meet different task requirements, improve performance, and adapt to other data formats.

Key words: Email chains detection, data mining, process extraction, process automation, text embeddings, sentence embedding, dinamic time wrapping, density-based clustering.

Contents

1 Introduction

1.1 Motivation

In the contemporary work environment, people are overwhelmed with the large amount of data that must be carefully managed and processed. This data is collected for various business, safety, and legal reasons, making manual handling impractical. Employees often end up doing repetitive tasks like reading, responding, attaching files, forwarding information, and managing spam, which lowers productivity and job satisfaction.

A 2017 study by McKinsey & Company [20] showed that nearly 60% of jobs could automate 30% or more of their activities with current technologies. Over time, we develop more tools that can modernize our daily work activities.

Automation offers a big solution to the challenges at work by significantly reducing costs and improving efficiency. By using automation technologies, businesses can offload routine tasks, allowing employees to focus on more strategic and innovative activities. Recent studies highlight the significant impact of automation on the workplace. For example, a McKinsey report [3] emphasizes the potential for automation to reshape various sectors by improving analytics and fostering human-machine collaboration.

Furthermore, a 2023 study by Deloitte [5] identifies key trends in workflow automation that are reshaping the future, showing how automation can streamline processes and enhance business performance. Another report by Smartbridge [32] notes that CIOs are increasingly integrating automation into their strategies to reduce inefficiencies and improve customer experiences.

These findings emphasize the necessity of adopting automation to manage the data load and streamline business operations effectively. There are many fields where automation could be applied, with various techniques for each specific case. In this study, we focus on exploring the possibilities for the business correspondence processes automation, developing one possible pipeline for this scenario.

1.2 Objectives

The primary objective of this project is to utilize data mining techniques on email data to extract potential business processes that could later be used for automating repetitive email management tasks. Specifically, we aim to:

- Develop a systematic methodology to reconstruct email chains and convert them into structured data.
- Create a pipeline for processing and clustering email chains, employing recent technologies where applicable.
- Identify candidate processes and evaluate them in situations where no ground truth is available.
- Assess the feasibility of the entire process and each stage individually in terms of time and computational resources.

1.3 Limitations

While the proposed project holds significant promise, it is important to acknowledge its limitations:

- The initial focus on email data means that the findings and solutions may be biased toward the specific characteristics of email communications, especially the chain extraction process. Extending these methods to other types of data may require additional adaptations.
- The evaluation stage is complicated by the lack of labeled data, which must be created to properly assess the quality of the developed methodology.
- Implementing the automation solutions at scale in a real-world environment may pose challenges to the existing workflow. Additional computational resources and some method modifications may be required.

Despite these limitations, the project aims to provide a framework for email process extraction, with potential applications extending to other areas of data management and business operations.

2 State of the Art

In this section, we delve into the concepts and areas of interest within process extraction and email mining. We discuss ongoing trends, the technologies currently in use, and highlight some recent papers and the challenges faced in these fields. Additionally, we explore the works related to the Enron email dataset and examine email mining from a data mining perspective, which is particularly relevant to our project. This comprehensive overview aims to provide a deeper understanding of the current landscape.

2.1 Process Extraction

Process extraction involves the identification and extraction of process-related information from unstructured data sources such as documents, logs, and databases. This technique is important in fields like business process management, where understanding and optimizing workflows can lead to significant efficiency improvements.

Due to its popularity and importance, the field of process mining has given rise to specialized conferences and events, such as the International Conference on Process Mining (ICPM). These conferences provide a platform for researchers, practitioners, and industry experts to share the latest developments, tools, and techniques in process mining and extraction.

Recent interests in process extraction have been driven by the integration of machine learning and natural language processing techniques. There are some notable trends:

- **Automated workflow discovery** involves using deep learning models to automate the extraction of workflows from textual data, reducing the need for manual process documentation. An example of this approach is demonstrated in the paper [18] which presents a framework for extracting multiple events from a single sentence using syntactic shortcut arcs and attention-based graph convolution networks, enhancing information flow and capturing long-range dependencies.

- **Event log analysis** involves improving methods for extracting events from logs using unsupervised and semi-supervised learning to identify patterns and anomalies in processes. For example, in the paper [13] the authors propose DQNLog, a method utilizing deep reinforcement learning to improve log anomaly detection. DQNLog analyzes log data by combining labeled and large-scale unlabeled data, using a deep Q-network to identify known and unknown anomalies.

- **Multi-source data integration** includes combining data from various sources such as textual documents, databases, and logs to create a comprehensive view of business processes. This approach facilitates decision-making, analytics, and an understanding of operations. Effective data integration strategies must address challenges like data heterogeneity, interoperability, and stakeholder engagement to ensure seamless and efficient data consolidation [28].

- **Real-time process monitoring** leverages streaming data analytics to monitor and extract processes in real-time, enabling proactive decision-making. This approach ensures timely detection and response to operational changes or issues. For instance, the paper [37] demonstrates the utility of real-time data processing by integrating fine-grained event extraction in legal contexts. Their method, EGG, showcases how real-time extraction and analysis of events can improve the generation of court views, ultimately helping legal professionals make decisions efficiently.

2.2 Email Mining

Email mining refers to the extraction of useful information and patterns from email data. This entails analyzing the content, metadata, and communication patterns within emails to derive insights. It is applied in various domains such as customer service, legal discovery, and organizational behavior analysis.

Among email mining tasks, the most popular nowadays are:

- **Sentiment Analysis:** Utilizing NLP techniques to assess the sentiment conveyed in emails, which can help in understanding customer satisfaction or employee morale. Sentiment analysis involves classifying the email content as positive, negative, or neutral, and can also extend to more nuanced emotions. Techniques such as lexicon-based approaches, machine learning classifiers, and deep learning models are commonly used.
- **Topic Modeling:** Implementing algorithms to identify and categorize the main topics discussed in large volumes of email communications. Topic modeling helps in summarizing the content of emails, discovering hidden patterns, and organizing the emails into meaningful clusters.
- **Spam Detection and Filtering:** Developing more sophisticated models to detect and filter spam emails, leveraging machine learning techniques to improve accuracy and reduce false positives. Techniques include Bayesian filtering, support vector machines, and neural networks. The goal is to distinguish between legitimate emails and unwanted spam, ensuring important communications are not lost while minimizing the intrusion of spam.
- **Network Analysis:** Analyzing email communication patterns to map out social networks within organizations, identifying key influencers and collaboration bottlenecks. Network analysis involves constructing graphs where nodes represent individuals and edges represent email interactions. This analysis can reveal insights into the structure and dynamics of communication within an organization, such as identifying central figures, understanding community structures, and detecting anomalies in communication patterns.

Next, we discuss the work already done with the Enron email dataset and other email data, also providing examples of email mining trends mentioned in this part.

2.3 Enron Emails Dataset

The Enron Email Dataset is a valuable resource for studying email communication and organizational behavior because of its real-world origins and large volume of data. In 2004, Shetty and Adibi [31] analyzed this dataset, highlighting its importance for research in data mining, social network analysis, and machine learning. The dataset contains about 0.5 million emails from around 150 users, mostly senior management at Enron, collected during the investigation into the company's collapse. Their work showed how the dataset could reveal patterns and trends in corporate communication, helping to understand internal dynamics and decision-making processes in large organizations.

Building on this, Diesner, Frantz, and Carley [7] explored the dataset further, focusing on social network analysis. They looked at email exchange frequency, the formation of communication clusters, and identified key actors within the network. This research provided insights into organizational structure, emergent leadership, and the flow of information within a company.

Agarwal et al. [1] expanded this work by creating a detailed hierarchy of Enron's organizational structure. They manually extracted this hierarchy from organizational charts in PDF attachments within the email corpus, resulting in a dataset of 1,518 employees and 13,724 dominance pairs. They compared a simple social network analysis (SNA) approach, based on degree centrality, to natural language processing (NLP) methods for predicting hierarchical dominance. They found that the SNA approach outperformed NLP systems, showing the effectiveness of SNA in this context. This gold-standard hierarchy is available as a MongoDB database.

In another study, Diesner, Frantz, and Carley [8] examined internal communication at Enron during its crisis using social network analysis. They enhanced the original email corpus by adding full names, career histories, and additional email addresses, and by cleaning the data for accuracy. They transformed the data into directed graphs with DyNetML, capturing email exchange direction and frequency. They found that during the crisis, communication became more diverse and interconnected, with previously disconnected employees starting to communicate directly. These findings provide insights into organizational behavior during crises and help validate theories of organizational failure.

Kathuria, Mukhopadhyay, and Thakur [15] focused on clustering emails using unsupervised clustering algorithms to improve data segregation, topic modeling, spam detection, and network analysis. Using emails from the Enron corpus, they implemented k-means and hierarchical clustering algorithms. These methods were evaluated using cosine similarity metrics to measure cohesion scores. They found that hierarchical clustering achieved a higher cohesion score than k-means, indicating better semantic similarity within clusters. The study concludes that clustering can provide valuable insights into communication patterns and themes within large email datasets.

The paper "Email User Classification and Topic Modeling" [30] used BERT to enhance email content analysis. BERT was used for feature extraction, converting email text into vectorized features that capture contextual and semantic meanings. These BERT-generated embeddings were used to classify emails by their authors, helping to detect misuse of the email server. Pre-trained BERT models were fine-tuned on the email dataset to improve classification accuracy. Comparative analysis showed that BERT-based models outperformed traditional methods like TF-IDF and CountVectorizer in classifying emails by their authors. BERT's embeddings also improved clustering methods like DBSCAN and LDA for topic modeling by providing high-quality, contextual feature vectors.

Finally, the paper "Actionable Phrase Detection using NLP" by Magotra [19] explores extracting actionable task phrases from raw text using linguistic filters and transfer learning. The goal is to identify phrases implying specific actions, useful in contexts like emergency task detection, first aid instructions, and productivity tools like automatic ToDo list generators. Using the Enron Email Dataset, the research applied custom-designed linguistic filters to clean the text data. The model was trained with the Universal Sentence Encoder to classify phrases as actionable or not. Key implementations included filtering sentences based on action verbs, sentence length, object pronouns, and negation verbs. The final model, a layered sequential model with the Universal Sentence Encoder, achieved high accuracy, F1 score, precision, and recall.

2.4 Email Chains Extraction

Since one of the steps of our project involves extracting email chains from unstructured email data, it is useful to discuss similar works and the methods applied.

Wang et al. [35] developed a method to improve the analysis of email conversations for digital forensics. Their approach focuses on extracting email headers, filtering redundant messages, and maintaining accurate parent-child relationships, even without Message-ID. They use a topic-based heuristic to merge or split threads into coherent conversations. By parsing email data from multiple users and using algorithms to map and adjust message threads, they showed their method has a competitive performance in tracking conversations and maintaining parent-child relationships.

Yeh and Harnly [36] proposed methods to reassemble email threads by establishing parent-child relationships between messages. They offered two approaches: one using the "Thread-Index" header from the Microsoft Exchange Protocol and another based on string similarity metrics and heuristic algorithms for cases without header information. Their similarity matching method considers subject lines, timestamps, and sender/recipient relationships, and includes a strategy for recovering missing messages. Using the Enron email corpus, they showed their method is effective in reconstructing threads and identifying parent-child relationships when header information is missing.

Repke and Krestel [26] aimed to improve the extraction of structure from free-text email threads for tasks like fraud detection and decision-making. They proposed a system called Quagga, which uses neural networks to identify and classify different parts of email texts, such as headers, bodies, greetings, and signatures. They encoded email lines into low-dimensional embeddings using convolutional neural networks and classified these lines with a gated recurrent unit-conditional random field (GRU-CRF) model. Their methods, tested on the Enron email corpus and a new dataset from Apache mailing lists, showed significant improvements over traditional rule-based and machine learning approaches in email segmentation tasks.

2.5 Current Project

In our project, we aim to solve a problem involving both email chain extraction and subsequent process extraction. Rather than utilizing the most powerful models available, our goal is to construct a comprehensive pipeline that performs each step with good quality, within our time and resource constraints, yet remains effective. Based on the studies we discussed in this section and their results, we have decided to use a transformer-based embeddings model and density-based clustering algorithms. These methods have demonstrated their efficiency compared to other approaches in the works cited [30] and [15].

3 Theoretical Background

In this section, the theoretical aspects necessary for the implementation are explained to ensure the comprehensibility of this work. The rationale behind the choice of specific methods can be found in the Methodology and Implementation section 4.

3.1 Word and Sentence Embeddings

Embeddings are a form of dense vector representation of words, phrases, or sentences that capture semantic information. Traditional bag-of-words or TF-IDF representations are sparse and high-dimensional, which can lead to computational inefficiency and increased storage requirements. Moreover, they do not capture semantic relationships between words, resulting in poorer overall performance.

Embeddings map textual data into continuous vector spaces where semantically similar items are closer together. This representation facilitates various natural language processing tasks, such as text classification, clustering, and similarity measurement since machine learning models work with numbers rather than texts.

Many algorithms can create embeddings, and the choice depends on the specific task, requirements, and resources available. It is essential to consider that more sophisticated models, despite their superior performance, may require more time and computational power. For example, in the study [34], the authors compare performance depending on the embeddings chosen for text classification. This is relevant because text classification requires a good capture of semantic meaning, which is also crucial for our project. The experiments are conducted for both context-dependent and context-independent models, exploring RNN-based and transformer-based structures. The performance proved to be better for transformer models, which we decided to use in our project.

Before diving deeper into transformer-based models, let's first cover some basics of other models to provide a clearer understanding of the background.

3.1.1 Word Embeddings

Word embeddings, such as Word2Vec [21], GloVe [22], and FastText [14], represent individual words based on their context within a corpus. These models learn to place words with similar contexts near each other in the vector space. However, these embeddings are context-independent, meaning they assign a single static vector to each word regardless of its usage in different sentences. This limitation means that word embeddings may not fully capture the nuanced meanings that words can take in various contexts.

Context-dependent models, either based on Recurrent Neural Networks (RNNs) or transformers, address the issue by considering the surrounding words and the overall sentence structure. RNN-based models were an earlier approach, and by now transformer-based models have shown superior performance.

BERT [6], a popular transformer-based model, has demonstrated exceptional performance in capturing the context and meaning of words within sentences. Its success is explained by its bidirectional attention mechanism, which allows it to understand context from both left and right directions, and its ability to pre-train on large corpora and then fine-tune on specific tasks. Despite these great results, BERT deals with token-level embeddings, which is not appropriate for our case when the embedding should represent a sentence or a short text.

3.1.2 Sentence Embeddings

To overcome the limitations of word embeddings, sentence embeddings represent entire sentences as fixed-size vectors. Sentence-BERT (SBERT) [25] extends the BERT capability to sentence-level semantics, efficiently generating semantically meaningful sentence embeddings.

3.1.3 Sentence-BERT

Sentence-BERT modifies the BERT architecture to produce sentence embeddings that are more suitable for tasks requiring sentence-level semantics. SBERT uses a siamese network structure to encode sentences into fixed-size vectors, which can then be compared using cosine similarity.

SBERT achieves this by leveraging the powerful contextual representations learned by BERT during pre-training and adding a mean pooling operation over the token embeddings generated by BERT to generate a single fixed-size vector for each sentence. This way, SBERT overcomes the issue BERT faces in deriving independent sentence embeddings. The sentence embeddings can then be used for various tasks, including semantic similarities, which are essential for our project.

3.2 Dynamic Time Warping

Dynamic Time Warping (DTW) [2] is a similarity measure used to compare time series data, sequences of data points, and polygonal curves. It is widely used due to robustness and insensitivity to outliers.

Given two sequences $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_m)$, DTW finds the optimal alignment between these sequences by warping the time axis iteratively. The DTW distance between P and Q is the minimal cost required to align the two sequences.

DTW can be defined as:

$$\mathrm{DTW}(P, Q) = \min_{\pi \in \Pi} \sum_{(i,j) \in \pi} d(p_i, q_j)$$

where π is a warping path that satisfies boundary conditions, monotonicity, and step size constraints, and $d(p_i, q_j)$ is a distance measure between points p_i and q_j.

A warping path is a sequence $\pi = ((i_1, j_1), (i_2, j_2), \ldots, (i_T, j_T))$ where each index pair (i_k, j_k) maps a point p_{i_k} in P to a point q_{j_k} in Q. The path must satisfy the following conditions:

- **Boundary conditions**: $(i_1, j_1) = (1, 1)$ and $(i_T, j_T) = (n, m)$
- **Monotonicity**: $i_{k+1} \geq i_k$ and $j_{k+1} \geq j_k$
- **Step size**: $(i_{k+1} - i_k, j_{k+1} - j_k) \in \{(1, 0), (0, 1), (1, 1)\}$

The classic DTW algorithm (code example 1) uses dynamic programming and has a time and space complexity of $O(nm)$, where n and m are the lengths of the sequences P and Q, respectively.

- **Distance Matrix**:

$$D(i, j) = d(p_i, q_j)$$

- **Dynamic Programming Recurrence**:

$$\mathrm{DTW}(i,j) = d(p_i, q_j) + \min \begin{cases} \mathrm{DTW}(i-1,j) \\ \mathrm{DTW}(i,j-1) \\ \mathrm{DTW}(i-1,j-1) \end{cases}$$

- **Boundary Conditions**:

$$\mathrm{DTW}(1,1) = d(p_1, q_1)$$

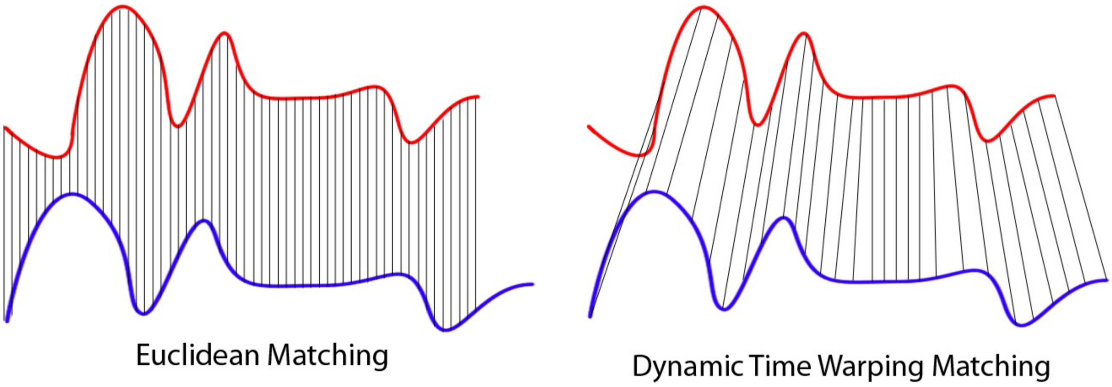


Figure 1. Comparison of Euclidean distance and DTW. Source: wikimedia.org

Algorithm 1 Dynamic Time Warping

```
Input: Sequences $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_m)$
Output: DTW distance between $P$ and $Q$
Initialize $DTW[0,0] \leftarrow 0$
for $i \leftarrow 1$ to $n$ do
    for $j \leftarrow 1$ to $m$ do
        $cost \leftarrow d(p_i, q_j)$
        $DTW[i,j] \leftarrow cost + \min(DTW[i-1,j], DTW[i,j-1], DTW[i-1,j-1])$
    end for
end for
return $DTW[n,m]$
```

3.3 Clustering Algorithms

Clustering is an important tool in data mining applications. Many clustering techniques have been proposed and implemented [11], achieving high-quality results in numerous domains.

However, many algorithms depend on the number of clusters initially provided, work only with data distributions of specific shapes, or cannot handle outliers, which limits their effectiveness in real-world scenarios.

These limitations are overcome by improved, efficient clustering techniques such as *DBSCAN (Density-Based Spatial Clustering of Applications with Noise)* and *HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)*. These algorithms can identify the number of clusters themselves, work with arbitrary shapes, and handle noise effectively.

3.3.1 DBSCAN

DBSCAN [10] forms clusters based on the density of data points in the feature space. This capability allows it to identify clusters of arbitrary shapes. The algorithm is robust to outliers as it filters out noise points that lie in low-density regions. However, it is important to note that the choice of parameters can significantly affect the results. Despite the DBSCAN advantages, it can struggle with datasets that have varying densities.

The algorithm performs the following steps:

- **Input:** DBSCAN takes as input a dataset D (feature or distance matrix), a neighborhood radius ε, and a minimum number of points $MinPts$ required to form a dense region.
- **Initialization:** Mark all points as unvisited.
- **Core Point Identification:** Randomly select an unvisited point p, mark it as visited, and check if its ε-neighborhood contains at least $MinPts$ points. If so, p is considered to be a *core* point; otherwise, it is marked as *noise*.
- **Cluster Formation:** If p is a *core* point, create a new cluster C and add p to C. Initialize a list of neighboring points N from p's ε-neighborhood.
- **Expansion:** For each point in N, if the point is unvisited, mark it as visited, add it to the cluster C and add its ε-neighborhood points to N if it has at least $MinPts$ points. A point is a *border* point if it is not a *core* point but falls within the ε-neighborhood of a *core* point (see fig. 2).
- **Iteration:** Repeat the expansion step until all points in N are processed. Then, move to the next unvisited point and repeat the process.
- **Completion:** The algorithm terminates when all points have been visited, resulting in a set of clusters, with some points potentially marked as noise.

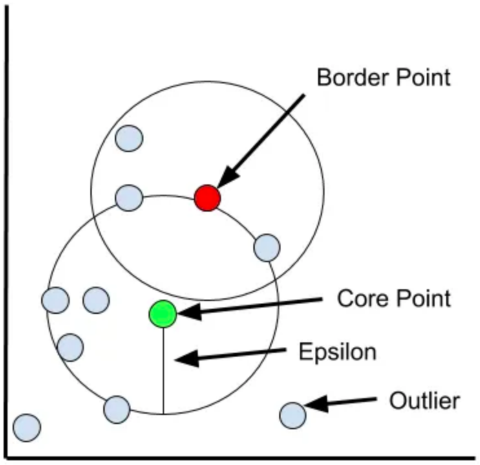


Figure 2. Illustration of DBSCAN clustering. Image source: towardsdatascience.com

The time complexity of DBSCAN depends significantly on how the region queries (finding neighboring points within a distance ϵ) are implemented.

In the naive implementation of the Region-Query, a linear scan of all the data points is used, leading to a time complexity of $O(n^2)$, where n is the number of data points. This is because each point is potentially compared with every other point in the dataset.

However, the region query operation can be significantly optimized using specialized data structures which can reduce the average time complexity to $O(n \log n)$ under certain conditions [29].

3.3.2 HDBSCAN

HDBSCAN [4] extends DBSCAN by converting it into a hierarchical clustering algorithm. HDBSCAN does not require the parameter ε, making it more adaptive to datasets with varying densities. The algorithm also includes mechanisms for soft clustering, allowing points to belong to multiple clusters with varying degrees of membership.

- **Input:** HDBSCAN takes as input a dataset D (feature matrix or distance matrix), a minimum cluster size *min_cluster_size*, and a parameter *min_samples* (similar to *MinPts* in DBSCAN) which affects the computation of core distances.

- **Transform the Space:** HDBSCAN begins by transforming the input space to a new space where density can be measured. This involves calculating the core distance for each point, which is the distance to its k-th nearest neighbor. The core distance $\mathrm{core}_k(p)$ of a point p is defined as:

  $$\mathrm{core}_k(p) = d(p, k\text{-th nearest neighbor of } p)$$

  To place two points in the same cluster, they must be in a sufficiently dense region and close to each other. This is where the mutual reachability distance $d_{\text{mreach-}k}(a, b)$ comes in, defined for any two points a and b as:

  $$d_{\text{mreach-}k}(a, b) = \max\left(\mathrm{core}_k(a), \mathrm{core}_k(b), d(a, b)\right)$$

  where $d(a, b)$ is the original distance between points a and b.

- **Build the Minimum Spanning Tree:** Using the mutual reachability distance, a minimum spanning tree (MST) of the points is constructed. Conceptually, it comes to considering the data as a weighted graph with the data points as vertices and an edge between any two points with a weight equal to the mutual reachability distance of those points.

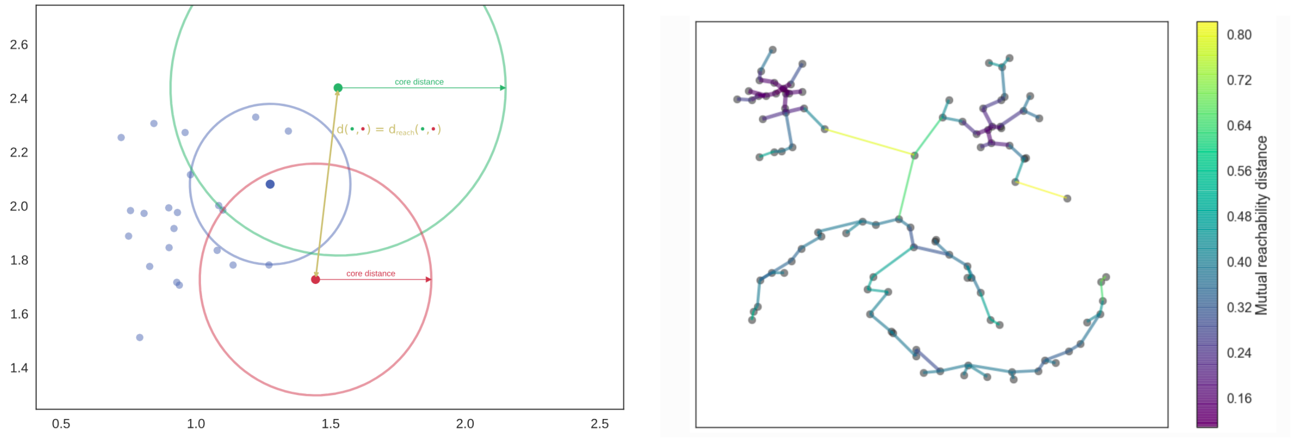
- **Build the Hierarchy of Clusters:** With the minimum spanning tree, the next step is to consider a threshold for the weights and remove any edges with weights above that threshold. As edges are removed, the graph starts to disconnect into connected components, forming clusters. By performing this for different threshold values, we obtain a hierarchy of connected components (from fully connected to fully disconnected). This hierarchical process, similar to aglomerative clustering, results in a dendrogram representing the nested clusters at different distance levels.

- **Condense the Cluster Tree:** The cluster hierarchy is then condensed into a smaller tree representing only the significant clusters. This step involves removing small clusters and retaining larger, more stable ones by applying a minimum cluster size constraint. A cluster is retained if it contains more points than *min_cluster_size*.

- **Extract Stable Clusters:** Finally, clusters are extracted based on their stability, a measure of how long they persist as the distance threshold varies. The stability of a cluster C is given by:

$$\text{Stability}(C) = \sum_{p \in \text{cluster}} (\lambda_p - \lambda_{\text{birth}})$$

  where $\lambda_{\text{birth}}(C)$ and $\lambda_{\text{death}}(C)$ are the birth and death levels of cluster C in the dendrogram (inversely proportional to distances). λ_p is the lambda value at which point p fell out of the cluster, which is between λ_{birth} and λ_{death}. The most stable clusters are chosen.



(a) Mutual reachability distance. **(b)** Minimum spanning tree.

Figure 3. Illustration of HDBSCAN clustering. Image source: hdbscan.readthedocs.io

The time complexity of HDBSCAN is $O(n \log n)$ for the construction of the minimum spanning tree and $O(n^2)$ for the cluster extraction process. Therefore, the overall complexity is $O(n^2)$. This is primarily due to the need to evaluate the stability of each possible cluster, which involves significant computational overhead.

Both DBSCAN and HDBSCAN are powerful tools for clustering, especially in cases where the data does not meet the assumptions of more traditional clustering methods such as k-means. These methods are particularly useful during the exploration stage when we are not aware of the number of clusters in advance and when the clusters may not be spherical in shape. DBSCAN's simplicity and effectiveness make it a common choice, while HDBSCAN's flexibility and robustness to varying densities provide a more advanced, albeit computationally more expensive, alternative for more complex data.

3.4 Evaluation Metrics

Different performance metrics are used to assess clustering results for various purposes: comparing clustering algorithms, evaluating two sets of clusters, and determining which of two clusters is better in terms of compactness and connectedness.

Generally, cluster validity measures are categorized into three classes:

- **Internal cluster validation**: based on the data clustered itself (internal information) without reference to external information.

- **External cluster validation**: based on some externally known result, such as externally provided class labels.
- **Relative cluster validation**: the clustering results are evaluated by varying different parameters for the same algorithm.

In our use case, we do not have the ground truth labels, so metrics suitable for internal cluster validation and relative validation should be used.

3.4.1 Silhouette Score

The Silhouette Score [27] is a measure of how similar an object is to its own cluster compared to other clusters. It ranges from -1 to 1, where a high value indicates that the object is well-matched to its own cluster and poorly matched to neighboring clusters.

For a data point i:

- $a(i)$ is the average distance between i and all other points in the same cluster (mean intra-cluster distance).
- $b(i)$ is the minimum average distance from i to all points in any other cluster (mean nearest-cluster distance).

The Silhouette Score for the data point i is then defined as:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

The overall Silhouette Score for a dataset is the average of $s(i)$ for all data points i.

3.4.2 Dunn Index

The Dunn Index [9] is a metric for evaluating the compactness and separation of clusters. A higher Dunn Index indicates better clustering, with more compact clusters are well-separated from each other.

The Dunn Index is defined as the ratio between the minimum inter-cluster distance to the maximum intra-cluster distance:

$$D = \frac{\min\limits_{1 \le i < j \le k} \delta(C_i, C_j)}{\max\limits_{1 \le i \le k} \Delta(C_i)}$$

where:

- $\delta(C_i, C_j) = \min\{d(x, y) : x \in C_i, y \in C_j\}$ is the distance between clusters C_i and C_j. This is the minimum distance between any two points in the two clusters.
- $\Delta(C_i) = \max\{d(x, y) : x, y \in C_i\}$ is the diameter of cluster C_i. This is the maximum distance between any two points within the same cluster.

4 Methodology and Implementation

This section provides a detailed overview of the methodology and implementation steps undertaken in this project. It begins by describing the dataset utilized and the preprocessing steps applied to clean and prepare the data for analysis. The subsequent sections delve into the process of restoring email chains from the dataset, highlighting the challenges faced and the solutions implemented to overcome them. The approach for obtaining numerical text representations is then detailed, followed by a distance matrix calculation. Finally, the clustering methods employed, are explained along with the strategies for parameter optimization and evaluation of the clustering results. The whole pipeline is showed in Figure 4 .

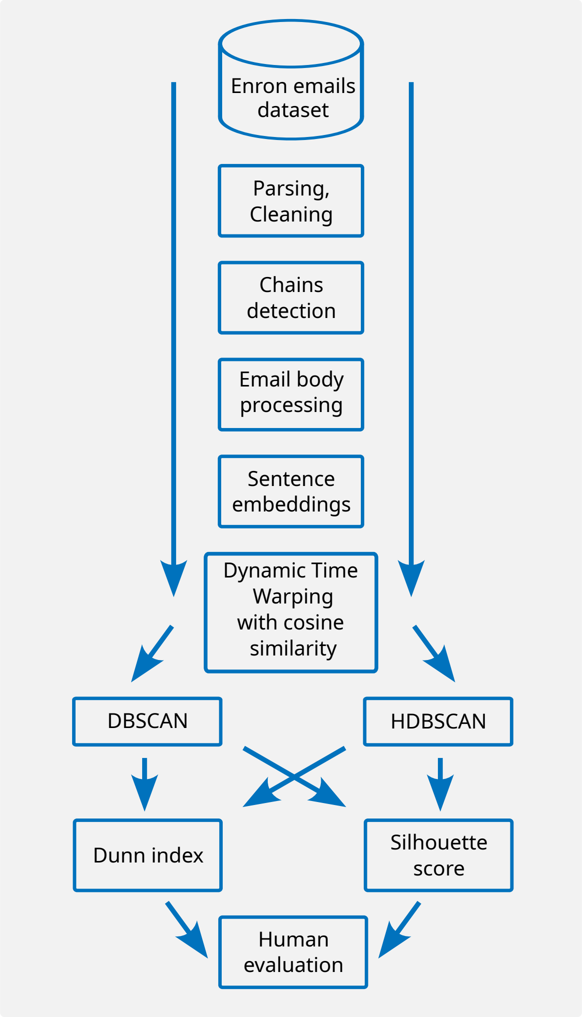


Figure 4. Project pipeline.

4.1 Data Description

The dataset utilized in this project is a curated version of the Enron corpus, specifically the CALO Enron Email Dataset. The original Enron corpus [16], made public during the legal investigations of Enron, contains over 600,000 emails from 158 users.

The CALO Enron Email Dataset [23], provided by the CALO Project, further enhances the original corpus by standardizing email attributes and removing attachments for consistency.

This version includes 517 401 emails from 150 senior managers at Enron, with improvements such as canonicalized dates and new Message-IDs to facilitate research. The dataset serves as a comprehensive resource, enabling the exploration of automated email management solutions through the analysis of diverse email structures and interactions.

In Figure 5, the main components of a typical Enron email are presented, which include metadata, email bodies, and despite the attachments being removed, the list of attachments for each email is still available. It's worth mentioning that there is no "InReplyTo" metadata, which makes the task of restoring email chains more difficult since we lack explicit links between replies and answers. Additionally, many emails contain the bodies of previous messages within the email bodies.

In the Appendix .1, there are some plots related to the exploratory data analysis of the Enron dataset.

4.2 Data Preprocessing

Using the initial Enron dataset, which is represented as a text file for each email grouped by type (inbox, sent items, deleted, etc.) and stored in the user's folder, the first goal is to develop a method for parsing, preprocessing, and grouping to restore the initial email chains. This section describes the first two aspects.

After the parsing stage, the dataset is represented as a .csv file with all the information extracted into separate columns such as: Message-ID, Date, Sender, Recipients, Subject, Email Body, etc. Next, the preprocessing part is responsible for:

- Cleaning email bodies to remove awkward symbols due to coding differences;
- Removing raw HTML emails;
- Removing emails with empty bodies;
- Extracting the email subject without "Re" or "Fwd" prefixes;
- Creating columns to indicate whether the email contains "Re" or "Fwd" prefixes;
- Converting dates to timestamps;
- Creating sets of recipients instead of using strings.

Also, at this point, duplicate handling is performed. While exploring the duplicates, it was noticed that some identical emails have different date information, which could be explained by errors during the extraction process from the original database. In Figure 6, the distribution of time differences between such duplicates is plotted. It is evident that the differences often amount to an integer number of hours, suggesting a mix-up between AM and PM notations. In the dataset, there are approximately 6.5 thousand such duplicates with time errors, which were removed along with other duplicates after carefully considering this issue. An example of this situation is shown in Figure 19 in the Appendix .2 for convenience.

After removing duplicates, we have 248 674 emails remaining. Additionally, we needed to delete all messages with missing "To" data since recipient information is important for chain extraction. Furthermore, we decided to discard 16 921 emails with empty subjects because they would adversely affect the quality of the chains. In the end, we have 222 325 emails in our dataset.

```
1 Message-ID:                  <31954872.1075856178745.JavaMail.evans@thyme>
  Date:                        Mon, 7 May 2001 08:39:00 -0700 (PDT)
  From:                        vince.kaminski@enron.com
  To:                          ron.baker@enron.com
  Subject:                     Internship Opportunities
  Mime-Version:                1.0
  Content-Type:                text/plain; charset=us-ascii
  Content-Transfer-Encoding:   7bit
2 X-From:                      Vince J Kaminski
  X-To:                        Ron Baker
  X-cc:
  X-bcc:
  X-Folder:                    Vincent_Kaminski_Jun2001_1/Notes Folders
                               /All documents
  X-Origin:                    Kaminski-V
  X-FileName:                  vkamins.nsf
3 Ron,
  The resume of the Rice student I mentioned to you.
  Vince
  ----- Forwarded by Vince J Kaminski/HOU/ECT on 05/07/2001 03:39 PM -----
4 "Ivy Ghose" <ghosei@ruf.rice.edu> on 05/03/2001 12:08:47 PM
  Please respond to <ghosei@rice.edu>
  To:  <vince.j.kaminski@enron.com>
  cc:  "Kenneth Parkhill" <Kenneth.Parkhill@enron.com>
  Subject:  Internship Opportunities
  Dear Mr.  Kaminski,
  I have found the EnronOnline project a very interesting one and have
  enjoyed working with everyone in the Research department as well as
  those from other departments.  I am keenly interested in this area and
  was wondering if there would be any summer internship opportunities.
  I have attached my resume to this mail for your review and look forward
  to hearing from you soon.
  Thank you
  IVY GHOSE
  RICE MBA 2002
5 - resume.doc
```

Figure 5. Parts of an Enron email: (1) Metadata, (2) Computer-generated metadata, (3) Body of the actual message, (4) Information from previous messages, (5) Attached documents.

4.3 Chains Detection

The first step in restoring email chains is to combine all the emails into subject groups, which consist of emails with the same subject. Each subject group may eventually lead to several distinct email chains. Emails within a subject group are ordered according to their timestamps, which are determined during the preprocessing stage.

The next step is to detect chains within each subject group. The main idea here is to rely

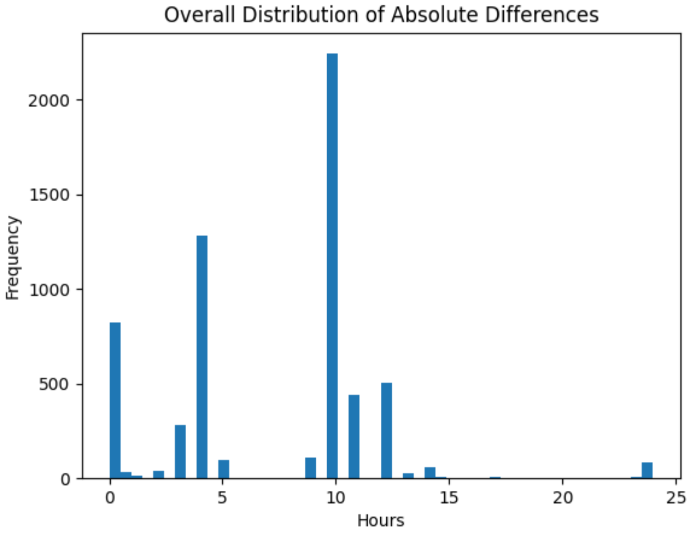


Figure 6. Distribution of Time Differences Between Duplicate Emails

on the subject, sender, recipients, and timestamp information of the emails, along with "Re" and "Fwd" markers in the subjects. This data is sufficient for an intuitive approach to chain detection. However, several issues were encountered in the original data that influenced the later process. These issues are listed below, with examples provided in the Appendix section .2 for better illustration:

- It would be reasonable to assume that the first email of a chain does not have "Re" or "Fwd" in the subject, and that subsequent emails do have these prefixes. However, this is not always the case. There are two main explanations for this: sometimes the very first email of the chain is missing from the data, or the prefix does not appear or may be manually removed by the user. (fig. 20)
- Since the emails were collected from only a portion of all Enron employees, sometimes emails in a chain are skipped. This can make it appear as though a person is communicating with themselves. I decided to keep these chains since this is a common pattern, and excluding them would lead to a significant loss of information for future steps. (fig. 21)
- Time errors persist in this step as well. Occasionally, the order of emails (when sorted by timestamps) is disrupted, which requires additional checks during the email chain detection process. (fig. 22)
- Some users have multiple email addresses, making it difficult to detect chains when a user uses different addresses. This is problematic because the email address is the primary identifier for a user. (fig. 23)

Considering the aspects mentioned above, the process of chain detection unfolds as follows:

We analyze each subject group separately, taking the first email as a potential starting point of the chain (emails are ordered according to the timestamps from the subject group creation step). This email is then compared to the next one, based on the conditions described below. If it meets the conditions, it is considered the next email in the chain. Subsequent emails are then compared to all emails already added to the chain. This means that even if the initial structure of the chain resembled a tree, it will be flattened according to the timeline.

After evaluating all emails in the subject group for their inclusion in the chain, there may be some emails left unassigned. The process is repeated for these remaining emails, and continues until every email in the subject group is assigned to a specific chain. This can include chains of just one email if no replies are present in the data.

Given the issues in the data, in addition to the subject, sender, recipients, and timestamp information, I also consider the email bodies. To address this, initial cleaning was performed, such as removing certain characters and patterns, attachment information (using regular expressions), and lowercasing the entire text.

When comparing two emails to determine if they belong to the same chain, the process follows these rules:

- **Both emails lack "Re" or "Fwd" prefixes in the subject:** In this case, to ensure they are part of the same chain, one email should contain the body of the other. This approach covers replies or forwards and accounts for time errors by checking in both directions (the first email contains the second, and vice versa).
- **The second email has "Re" or "Fwd" prefixes in the subject:** Here, we check if the second email contains the body of the first one or if the sender of the second email is one of the recipients of the first email. This method covers forwarding, replying, and follow-up emails.
- **The first email has "Re" or "Fwd" prefixes in the subject, while the second does not:** This could indicate a time error. To verify this, we apply the same conditions as in the previous case, but swap the positions of the first and second emails.

In all cases, we set a rule that the maximum time gap between two emails should be three months. This prevents emails from similar chains, sent by the same people in different periods, from being incorrectly grouped together.

After executing the described process, we restore the chains. The length distribution of these chains is presented in Table 1, along with the subject group size distribution. For subsequent steps, we retain only the chains with a length of 2 or more (a total of 28 907 chains, which corresponds to 79 805 emails) and discard the single emails (chains of length 1). The most popular email chains subject are presented in Figure 7.

4.4 Text Representation

Obtaining numerical text representation is the next necessary step in our project, for which we use sentence transformer embeddings as discussed in the theoretical section 3.1. This choice ensures the quality of text representation, as pretrained transformer models have demonstrated outstanding results.

Before embedding the emails from our data, it is important to carefully consider which parts of the emails should be used as input for the transformer model. The fact is that emails often include text from previous messages to which the user replied or forwarded. We cannot retain all this information, as it would make the representations for emails from the same chain similar, due to the inclusion of the same text. Therefore, we conclude that it is necessary to remove the information from previous messages in the current one. The subject of the email and information about possible attachment files are also included in the email text prepared for the embedding process. This results in a significant reduction in the number of words in the emails, which can be observed in Figure 8.

Chains		Subject Groups	
Length	Number of Instances	Size	Number of Instances
1	141 910	1	86 056
2	18 354	2	20 584
3	5 694	3	7 649
4	2 384	4	3 704
5	1 072	5	1 917
6	553	6	1 131
7	329	7	730
8	169	8	463
9	112	9	317
10	70	10	269
> 10	170	> 10	1 083
Total	170 817	Total	123 903

Table 1. Chains and Subject Groups Distribution

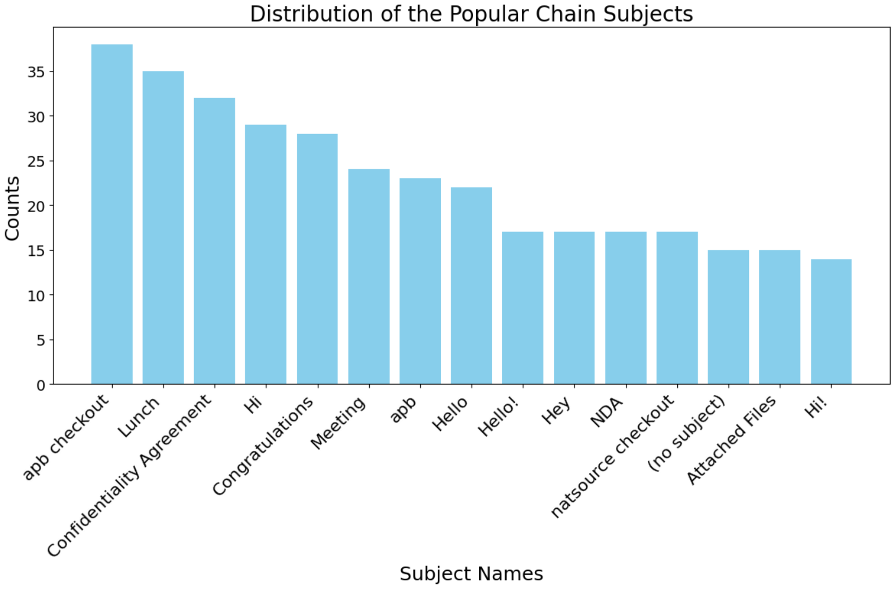
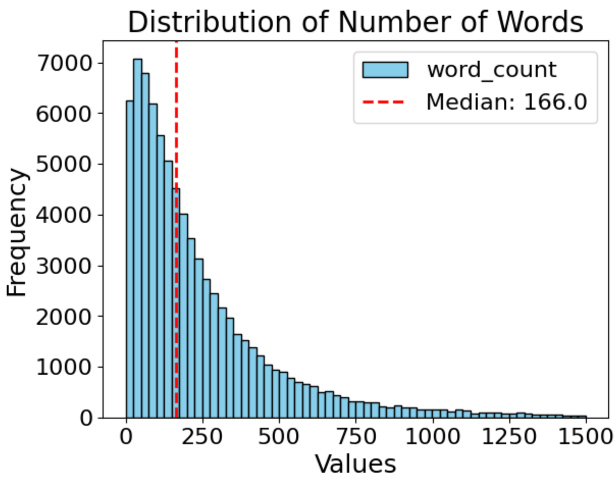


Figure 7. Popular email chains subjects.

To create embeddings, we use the all-mpnet-base-v1 model [12], which is part of the Sentence Transformers library [25]. This model is based on the MPNet (Masked and Permuted Pre-training) architecture [33], which enhances the BERT and RoBERTa models by integrating permuted language modeling with masked language modeling. The all-mpnet-base-v1 model excels at capturing semantic relationships between sentences, making it highly effective for our use case. It can handle sentences and short texts, taking at most 512 tokens, which is approximately equal to 300-400 words, appropriate for our emails after cleaning.

4.5 Distances

In this section, we describe our approach for calculating the distance matrix between embedded email chains. To compute the distances between email chains, we employ the Dynamic Time Warping algorithm, detailed in Subsection 3.2 of the theoretical section 3. This method is chosen for its suitability in measuring similarity between two temporal sequences of different



(a)

(b)

Figure 8. Distribution of the number of words in emails (a) before cleaning, (b) after cleaning.

lengths. For DTW, we use cosine similarity as the distance measure between individual emails in the chains.

The cosine similarity between two vectors **a** and **b** is defined as:

$$\text{cosine\_similarity}(\mathbf{a}, \mathbf{b}) = 1 - \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\| + \epsilon} \tag{1}$$

where $\mathbf{a} \cdot \mathbf{b}$ is the dot product of the vectors, $\|\mathbf{a}\|$ and $\|\mathbf{b}\|$ are their respective norms, and ϵ is a small value added to prevent division by zero. This choice of cosine similarity is motivated by its effectiveness in measuring the similarity between high-dimensional vectors, such as those produced by transformer embeddings.

DTW computes the optimal alignment between two sequences by minimizing the cumulative distance between them. Given two time series **A** and **B**, the DTW distance is computed as follows:

1. Initialize the cost matrix E with dimensions (l_1, l_2), where l_1 and l_2 are the lengths of **A** and **B**, respectively.
2. Fill the first cell with the cosine similarity between the first elements of **A** and **B**.
3. Fill the first column and first row by accumulating the cosine similarity values.
4. Fill the remaining cells with the minimum cumulative distance from the neighboring cells (top, left, and top-left).
5. The DTW distance is the value in the bottom-right cell of the cost matrix E.

To efficiently compute the DTW distance matrix for the large dataset (we have 28 907 email chains, which results in about 900 millions DWT runs, each of a time and space complexity of $O(nm)$, where n and m are the lengths of the two sequences), we leverage Numba, a Just-In-Time (JIT) compiler for Python [17]. Numba translates a subset of Python and NumPy code into fast machine code, allowing for significant performance improvements, especially in numerical computations. It also enables parallelized computation of the DTW distance matrix, drastically reducing computation time.

The results for the chain distances are represented in Figure 9.

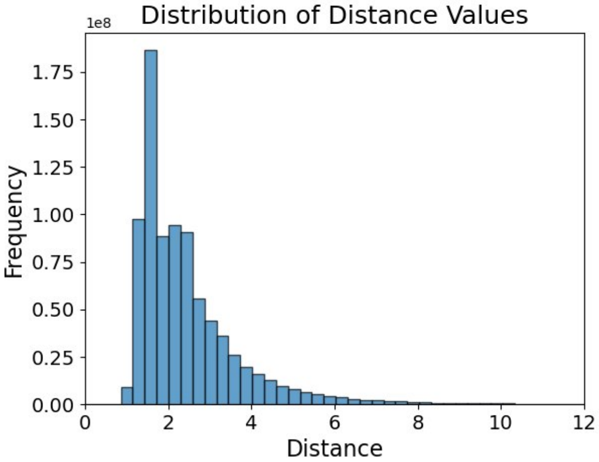


Figure 9. Chains distances distribution.

4.6 Clustering

The last step before the evaluation and analysis part is to perform clustering. We chose to use the density-based clustering algorithms DBSCAN and HDBSCAN, as detailed in subsection 3.3 of the theoretical part 3. To optimize these algorithms, we utilize a precomputed distance matrix discussed in 4.5. Given our high-dimensional text representations of 768 dimensions, calculating distances within the algorithms would be time-consuming. Although this optimization is beneficial, we lose some information by relying solely on distances and discarding the vector features, rendering some clustering algorithms and evaluation metrics unsuitable for the precomputed distance matrix.

Our selection of these algorithms is justified by their ability to work with densities and apply to any data shape distribution, unlike traditional k-means, which is optimal for spherical distributions. Since we do not know our data distribution, a density-based approach appears promising. Furthermore, these clustering methods can handle noise, which is crucial for our use case. We aim to identify potential processes among a large volume of emails, where there is a significant amount of correspondence that does not follow process patterns and is not suitable for automation. This irrelevant data is essentially noise, which we encounter more frequently than potential process chains. Generally, our email chains distribution includes small dense regions that clustering algorithms can identify due to their similarity, making them viable process candidates. Therefore, the ability to classify noise is higly important.

DBSCAN and HDBSCAN share a similar foundation, but HDBSCAN is more powerful as it can detect clusters of varying densities and offers a more flexible clustering structure. However, this enhanced capability makes HDBSCAN more computationally expensive and slower than DBSCAN. The increased computational time and resource usage in HDBSCAN arise from its complex algorithm, which involves building a hierarchy of clusters, maintaining a minimum spanning tree, and calculating cluster stability, requiring more extensive calculations and greater memory usage.

When performing clustering, choosing the appropriate parameters for the algorithms is crucial. Evaluation metrics play a key role in this process. Two metrics we describe in subsection 3.4

are the Silhouette score and the Dunn index. Additionally, there are other metrics such as the Davies-Bouldin score and the Calinski-Harabasz score. The Davies-Bouldin score considers the ratio of within-cluster scatter to between-cluster separation, while the Calinski-Harabasz score measures the ratio of the sum of between-cluster dispersion and within-cluster dispersion for all clusters. However, these metrics require centroids for computation, which we do not have since we use a distance matrix rather than feature vectors. Moreover, these two metrics do not perform well for density-based algorithms because they work better for a specific shape and distribution of clusters, typically spherical or convex shapes. Density-based algorithms, on the other hand, can produce clusters of arbitrary shape and varying density, which violates these assumptions and leads to poor performance of these metrics.

For parameter selection, we use grid search, which we describe separately for each algorithm below. Grid search requires a metric for evaluation. In this case, we chose the Silhouette score because it is faster than the Dunn index. The Dunn index is slower because it requires calculating the maximum inter-cluster distance and the minimum intra-cluster distance for all pairs of clusters, which is computationally intensive. The Silhouette score measures how similar an object is to its own cluster compared to other clusters. Practically, good values for the Silhouette score range from 0.5 to 1.0, indicating well-defined clusters. Values around 0 indicate overlapping clusters, and negative values suggest that samples might have been assigned to the wrong clusters.

4.6.1 DBSCAN

For the DBSCAN algorithm, we have two parameters to optimize:

- ε (Epsilon): This parameter defines the maximum distance between two points for them to be considered neighbors. It works this way because the core principle of DBSCAN is to find regions of high point density that are separated by regions of low point density. If ε is small, only points that are very close to each other will be considered part of the same neighborhood, resulting in small and potentially fragmented clusters, with more points classified as noise. Conversely, if ε is large, more points will fall within the same neighborhood, leading to larger clusters and fewer points classified as noise. This parameter directly influences the scale of the neighborhoods and thus the granularity of the clustering.

- $MinPts$ (Minimum Samples): This parameter specifies the minimum number of points required to form a core point. A core point is a point that has at least $MinPts$ points within its ε-neighborhood. This works as a way to define density: regions with a high density of points will have many core points, whereas regions with low density will have fewer or no core points. If $MinPts$ is set too low, almost every point may become a core point, leading to many small clusters. If $MinPts$ is set too high, fewer points will meet the criteria to become core points, which may result in fewer, larger clusters and more points being labeled as noise. This parameter helps to distinguish between noise and actual clusters based on density.

There is a method to determine the optimal ε parameter for DBSCAN [24], which involves using the k-nearest neighbors (k-NN) algorithm. First, compute the k-distance (distance to the k-th nearest neighbor) for each point in the dataset, where k is typically set to the value of $MinPts$. Then, plot these k-distances in ascending order, and look for a "knee" or an elbow in the plot; this point indicates a suitable value for ε, as it represents a distance threshold that distinguishes dense regions from sparser ones. Unfortunately, in our case, this method does not work, pointing out to an excessively large value for epsilon. This behavior can be explained by the significant amount of noise data, which is not usually assumed for the clustering task. Thus, in this unlucky situation, we proceed to use the grid search approach.

The Epsilon parameter was explored with values of 0.0001, 0.001, 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, and 0.8 (later the values 0.0001, 0.001, 0.01 turned out to be too small for our chain distances distribution). For each epsilon value, the minimum samples parameter was varied in the range from 2 to 100. The results for the Silhouette score are presented in Figure 10 .

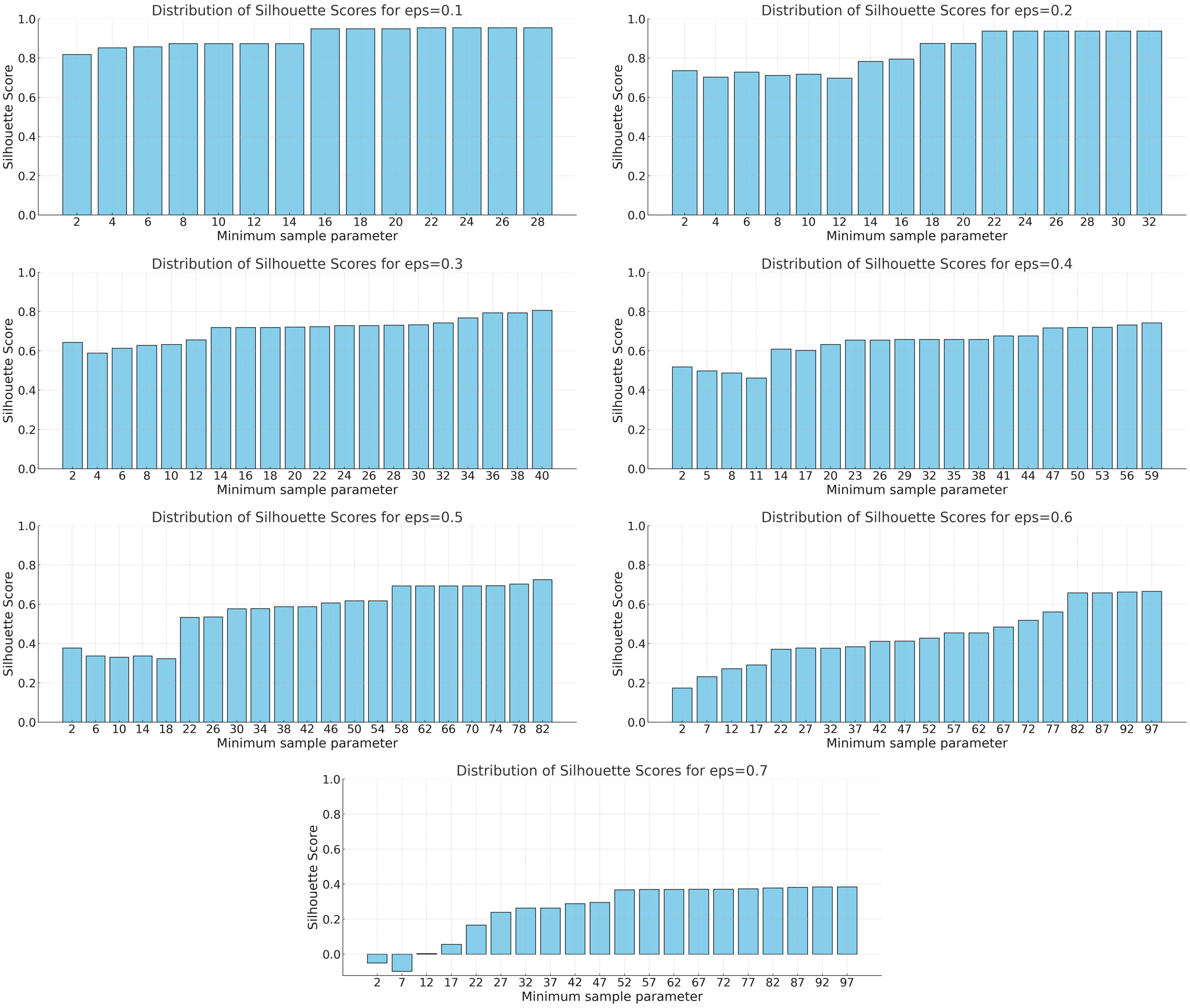


Figure 10. Distribution of Silhouette Scores for various ε values

Additionally, it is worth noting that with increasing values of $MinPts$ for each ε, the number of clusters decreased, and the average number of elements per cluster increased correspondingly.

4.6.2 HDBSCAN

HDBSCAN constructs a hierarchy of clusters by varying the density threshold, eliminating the need for a predefined epsilon parameter. This flexibility allows HDBSCAN to identify clusters with varying shapes and densities. Key adjustable parameters include:

- *min_samples*: This parameter sets the minimum number of points required to form a dense region. Increasing *min_samples* makes the algorithm more selective, resulting in fewer and larger clusters, as more points are classified as noise. This happens because a higher *min_samples* value raises the threshold for what constitutes a dense region, thus requiring more points to meet the criteria. Conversely, decreasing *min_samples* makes the algorithm less selective, allowing more points to form clusters. This leads to the detection of smaller and more numerous clusters, as even regions with fewer points can be considered dense enough to form a cluster.

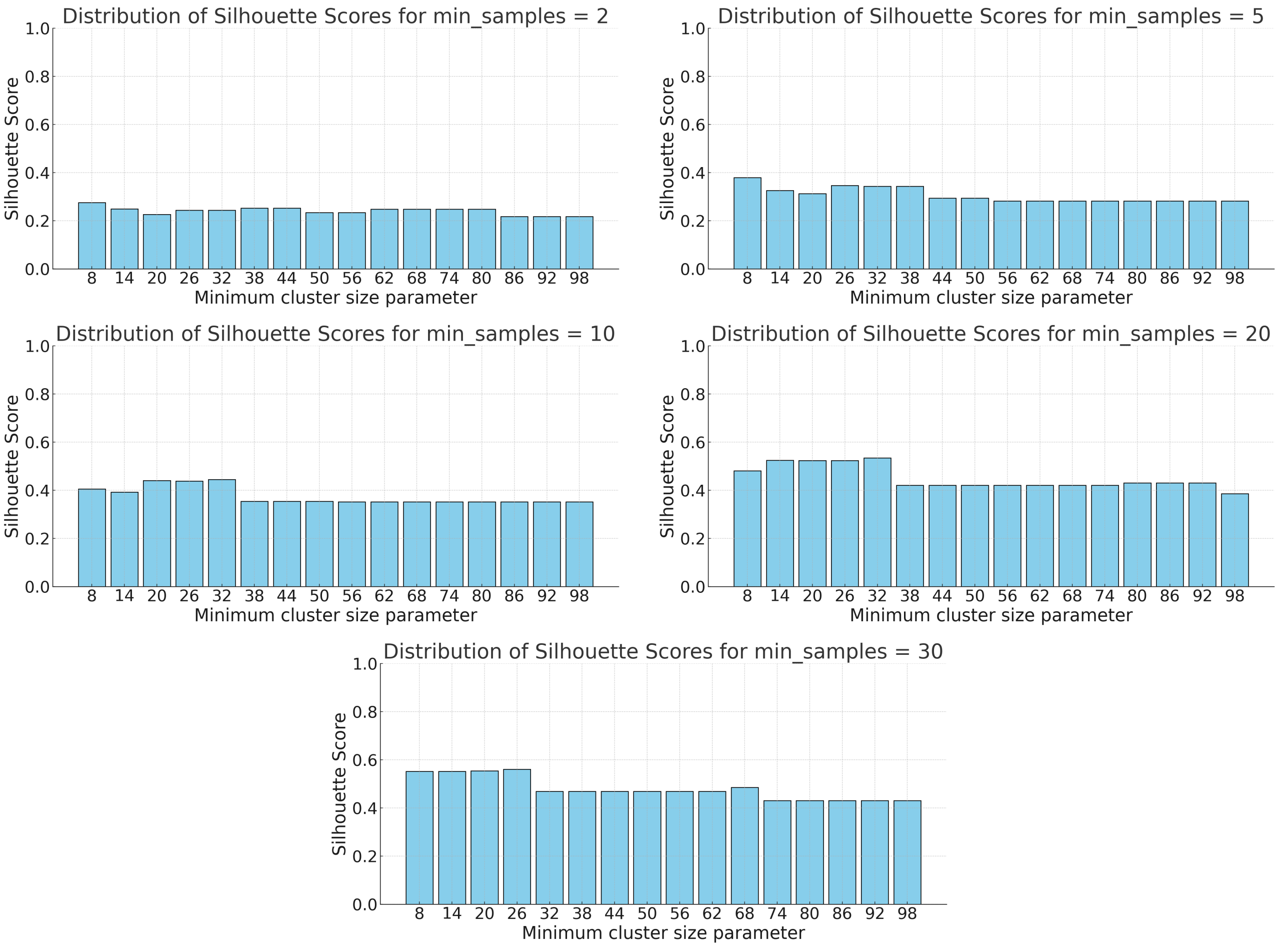
- *min_cluster_size*: This parameter determines the minimum size of clusters. Increasing *min_cluster_size* causes the algorithm to merge smaller clusters into larger ones and treat smaller groups of points as noise. This happens because clusters smaller than the specified size are considered insignificant and are either absorbed into larger clusters or classified as noise. Decreasing *min_cluster_size* allows for the detection of smaller clusters, providing finer granularity in the clustering results. This means that the algorithm can identify and retain smaller, more detailed structures within the data, as even small groups of points can form valid clusters.

The *min_samples* parameter was explored with values of 2, 5, 10, 20, and 30. For each *min_samples* value, we set *min_cluster_size* in the range from 2 to 100 with a step of 6 (*min_cluster_size* of 2 was discarded as it resulted in an excessive number of clusters, which is not realistic). We did not investigate larger values of *min_cluster_size* since our goal is to identify process candidates, and it is unlikely that more than 100 emails represent the same process, given that a large portion of the emails in the data do not pertain to business correspondence. The results of the parameter selection using the Silhouette score are presented in Figure 11 .

For each *min_samples*, as *min_cluster_size* increases, the number of clusters decreases, while the average number of elements in each cluster increases. This is important information for later evaluation.

For the assessment phase, we decided to proceed with 3 options for each clustering algorithm. Our selection was based on several criteria, including the Silhouette score (while acknowledging that it does not fully represent clustering quality) and the number of clusters obtained. Additionally, we considered the Silhouette score per cluster, ensuring it was not below zero (which would indicate incorrect clustering), and the subject names of the email chains within the clusters. Similar subject names within a cluster are a positive sign, as they may indicate a process. We also evaluated the average number of elements in each cluster, ensuring it was neither too small (fewer than five) nor too large (more than several hundred), as the processes we seek should be representative within the data. We are particularly interested in examining how different parameters influence the results, especially regarding the level of granularity for process candidates.

1. DBSCAN:
    - $\varepsilon = 0.4$, $MinPts = 40$
    - $\varepsilon = 0.5$, $MinPts = 5$
    - $\varepsilon = 0.3$, $MinPts = 12$
2. HDBSCAN:
    - $min\_samples = 10$, $min\_cluster\_size = 20$
    - $min\_samples = 10$, $min\_cluster\_size = 8$
    - $min\_samples = 30$, $min\_cluster\_size = 20$


Distribution of Silhouette Scores for min_samples = 2

Distribution of Silhouette Scores for min_samples = 5

Distribution of Silhouette Scores for min_samples = 10

Distribution of Silhouette Scores for min_samples = 20

Distribution of Silhouette Scores for min_samples = 30

Figure 11. Distribution of Silhouette Scores for various *min_samples* values

4.7 Evaluation Strategy

Evaluation of the results obtained is quite a subjective part, since we do not have a ground truth for clustering and an objective definition for a process in this case. Therefore, human evaluation plays an essential role in this part.

There are several ideas on how to evaluate and analyze the results:

- Calculate the Dunn Index and Silhouette score for the clustering results;
- Compute the Silhouette score per cluster and then manually check if there is a correlation between human evaluation and the metric, specifically if the clusters with higher Silhouette scores more closely resemble the process compared to those with lower scores;
- Observe the differences in granularity for different parameters within one algorithm. It could be that some processes are considered to be different for one set of parameters and are combined into the same cluster for another set of parameters;
- Estimate the proportion of potential processes by human evaluation;
- Compare DBSCAN and HDBSCAN results.

5 Results Analysis

In this section, we aim to assess the results obtained after the clustering stage using the evaluation strategies described in Section 4.7. We compare two clustering algorithms and analyze how their parameters influence the results, seeking insights for future use. Additionally, we manually evaluate the presence of process candidates among our clusters and verify if the metric correlates with human evaluation.

In Tables 2 and 3, each clustering result is evaluated using the Silhouette score and the Dunn index. The bar plots display the Silhouette score for each cluster and the cluster sizes. Here, we describe some clusters to give an impression of the results and justify the conclusions made. The cluster examples can be found in the Appendix, Section .3.

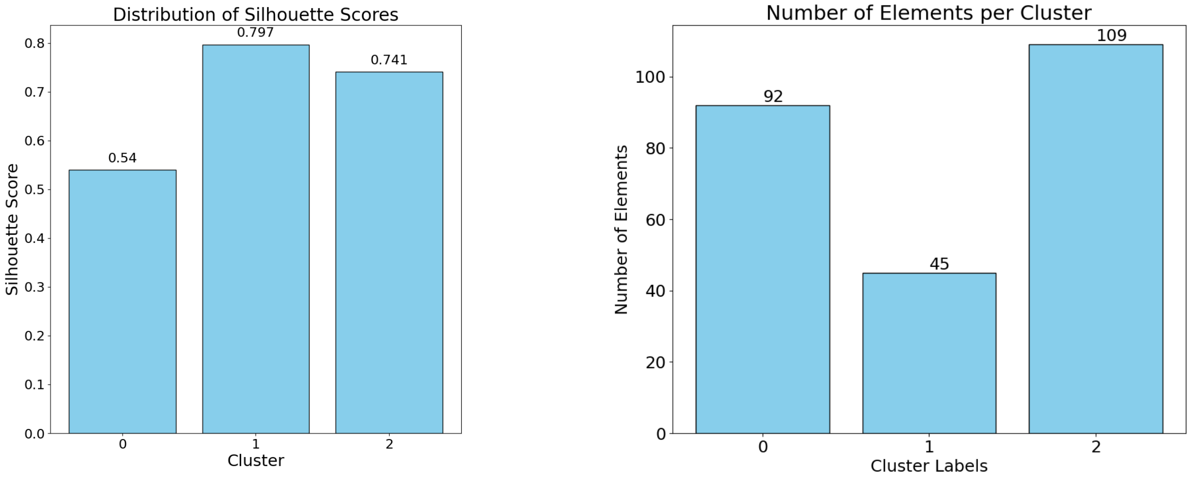
5.1 DBSCAN

ε	$MinPts$	number of clusters	Silhouette Score	Dunn Index
0.4	40	3	0.6762	0.3526
0.5	5	35	0.3247	0.1697
0.3	12	7	0.6562	0.4264

Table 2. DBSCAN Metrics.

1. $\varepsilon = 0.4$, $MinPts = 40$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 218 chains of the length 2, 28 chains of the length 3.



(a) Silhouette score per cluster. **(b)** Number of elements per cluster.

Figure 12. Results for DBSCAN $\varepsilon = 0.4$, $MinPts = 40$.

Cluster 0: This cluster consists entirely of "EOL Credit Responses" emails sent on various dates.

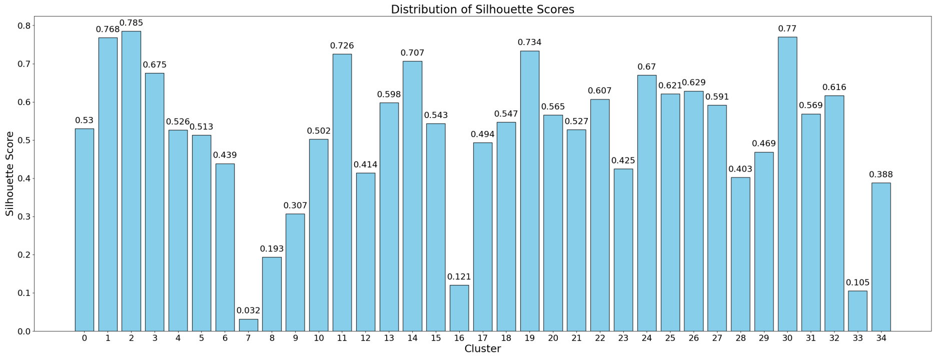
Cluster 1: This cluster contains "EOL Approvals" emails with the body text "please see attached."

Cluster 2: This cluster includes ERV/TRV Notification emails regarding the issuance of a new report. The clustering algorithm identified this as an already automated process.

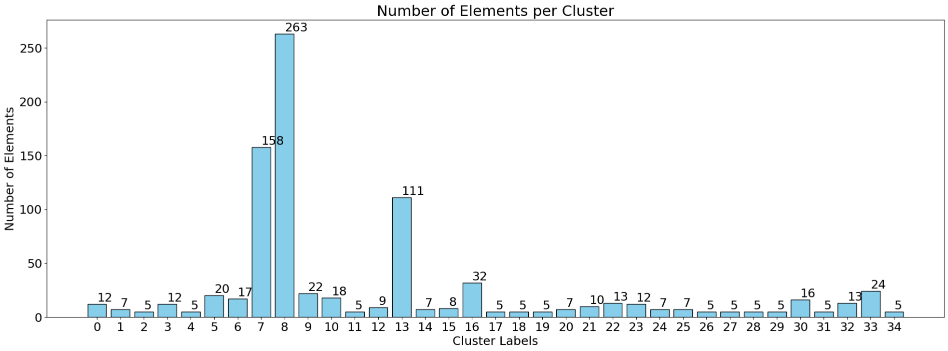
Clusters 0 and 1 represent processes that could potentially be automated, assuming they are not already automated, which is not clear from the email content.

2. $\varepsilon = 0.5$, $MinPts = 5$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 762 chains of the length 2, 89 chains of the length 3, 7 chains of the length 4, 5 chains of the length 5, 1 chain of the length 6, 1 chain of the length 7.



(a) Silhouette score per cluster.



(b) Number of elements per cluster.

Figure 13. Results for DBSCAN $\varepsilon = 0.5$, $MinPts = 5$.

Cluster 1: ClickPaper Approvals with the text "Please see attached." These emails are almost identical chains.

Cluster 2: Emails mentioning EDR Report Files for different quarters, forming almost identical chains.

Cluster 19: Global Government Affairs conference emails. The first email contains a template for the announcement, and the second discusses a schedule change in a free format.

Cluster 30: ENRON Performance Management Review Feedback using a template email.

Cluster 7: Emails containing a signature block similar to:

Debra Perlingiere
Enron North America Corp.
Legal Department
1400 Smith Street, EB 3885
Houston, Texas 77002
dperlin@enron.com
Phone 713-853-7658
Fax 713-646-3490

Many of these emails also contain agreements or confirmations.

Cluster 16: Energy-related emails involving selling, stocks, and investments.

Clusters 33 and 34: APB deals and checkouts. These clusters are good process candidates, as the emails vary from one another. They could potentially be combined into the same cluster.

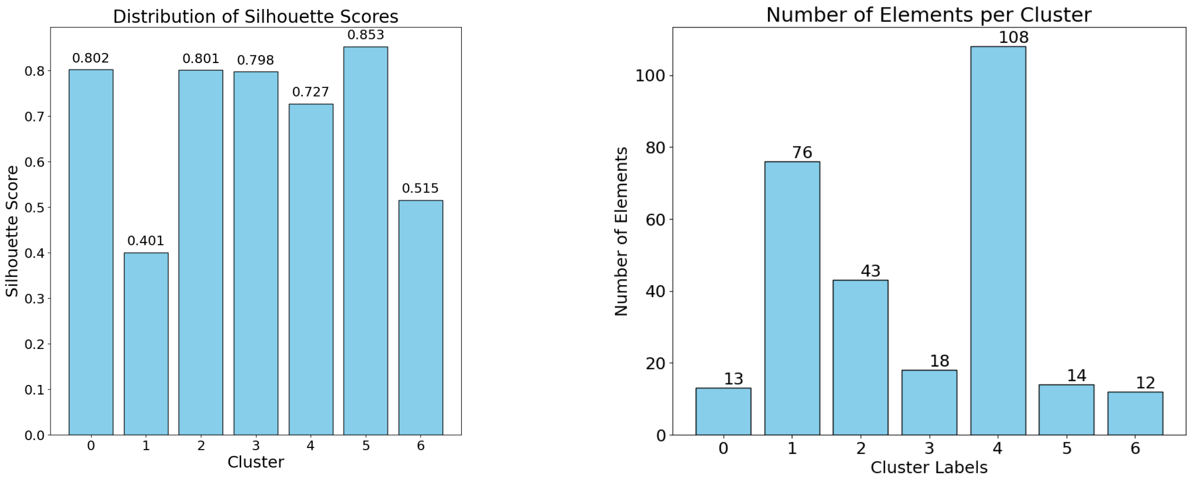
Cluster 25: Emails asking for and reminding about timesheet updates. This cluster is a good candidate for automation.

Cluster 28: Various congratulatory emails. These are not potential process candidates but contain similar sentiments.

By exploring this clustering, we conclude that our approach captures both similar sentiment chains and more automated, similarly structured chains. The score is much higher for the latter.

3. $\varepsilon = 0.3$, $MinPts = 12$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 242 chains of the length 2, 40 chains of the length 3, 1 chain of the length 6, 1 chain of the length 7.



(a) Silhouette score per cluster.

(b) Number of elements per cluster.

Figure 14. Results for DBSCAN $\varepsilon = 0.3$, $MinPts = 12$.

Cluster 0: ClickPaper approvals: "Attached are ClickPaper credit approvals for {date}"

Cluster 1: EOL Credit Responses.

Cluster 2: EOL Approvals.

Cluster 3: EOL Credit Responses: "Please find attached Credit's EOL responses for {date}"

Cluster 4: ERV/TRV Notifications.

Cluster 5: ENRON Performance Management Review technical emails.

Cluster 6: EOL Credit Responses. Tana's human commentary in the last email.

All chains in these clusters either have a template or follow a similar structure.

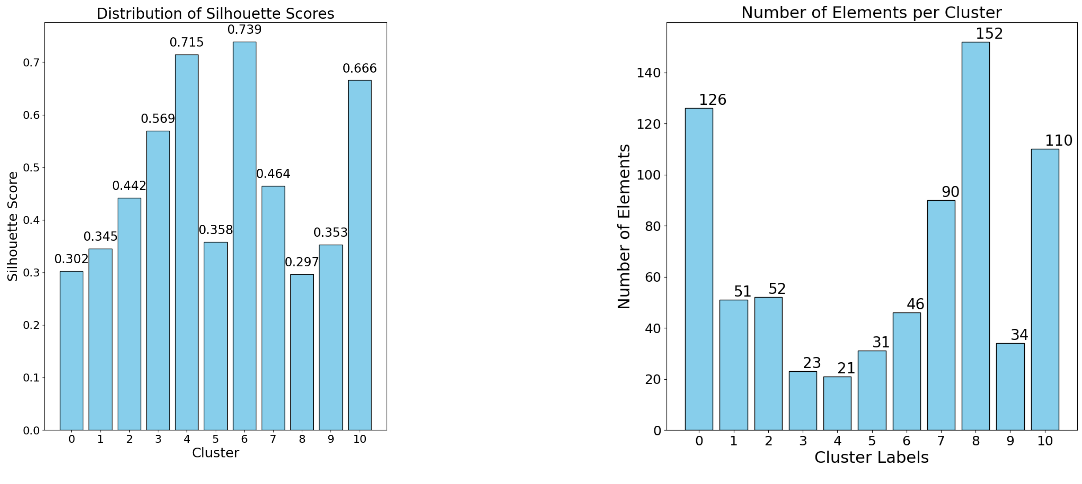
5.2 HDBSCAN

$min\_samples$	$min\_cluster\_size$	number of clusters	Silhouette Score	Dunn Index
10	20	11	0.4401	0.2203
10	8	28	0.4059	0.0269
30	20	6	0.5545	0.2760

Table 3. HDBSCAN Metrics.

1. $min\_samples = 10$ $min\_cluster\_size = 20$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 686 chains of the length 2, 48 chains of the length 3, 1 chain of the length 4, 1 chain of the length 5.



(a) Silhouette score per cluster.

(b) Number of elements per cluster.

Figure 15. Results for HDBSCAN $min\_samples = 10$ $min\_cluster\_size = 20$.

Cluster 0: Emails are short and all contain a 6-digit number. Most of them are related to APB checkouts, invoices, and bookings. The processes are similar, though somewhat scattered.

Cluster 1: Meeting emails with dates and times, potential for automation.

Cluster 2: Congratulations emails, mainly on promotions. They have a similar tone but lack process potential for automation.

Cluster 4: Automated emails: "Please find attached Credit's EOL responses for {date}."

Cluster 5: ClickPaper approval short emails with attachments, including the word "attached."

Cluster 6: Short emails with attachments, mainly stating "Please see attached."

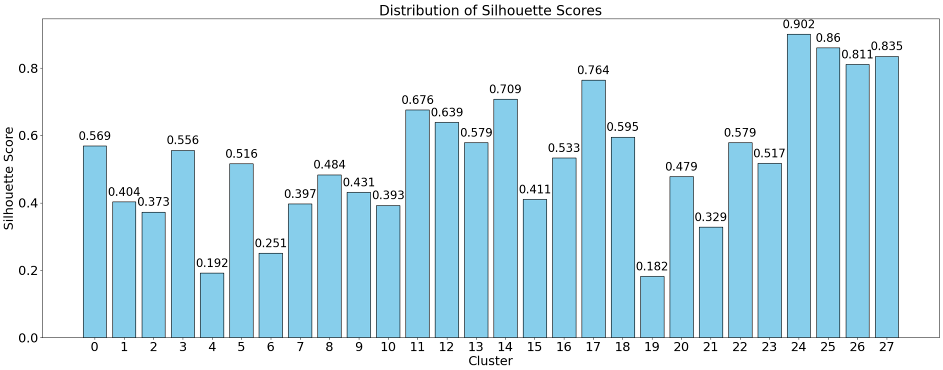
Cluster 7: Automated emails: "Please find attached Credit's EOL responses for {date}."

Cluster 10: ERV Notification automated emails. All email bodies follow the form: "The report named {report_name} {link}, published as of {date} is now available for viewing on the website."

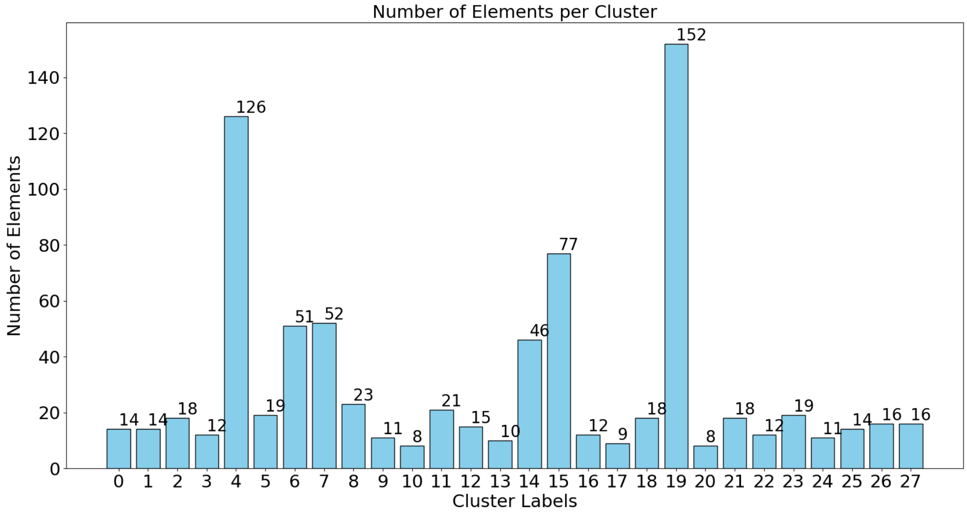
Clusters 3, 8, 9: Do not have specific distinct traits; they sometimes include agreements, but the email chains are quite varied.

2. $min\_samples = 10$ $min\_cluster\_size = 8$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 776 chains of the length 2, 42 chains of the length 3, 1 chain of the length 4, 1 chain of the length 5, 1 chain of the length 6, 1 chain of the length 7.



(a) Silhouette score per cluster.



(b) Number of elements per cluster.

Figure 16. Results for HDBSCAN $min\_samples = 10$ $min\_cluster\_size = 8$.

Cluster 22: TRV notifications about issuing a new report.

Cluster 23: ERV notifications about issuing a new report.

Cluster 24: ERV notifications about issuing the report 'Violation/Notification Memo'.

Cluster 25: ERV notifications about issuing the report 'Enron Americas Position Report'.

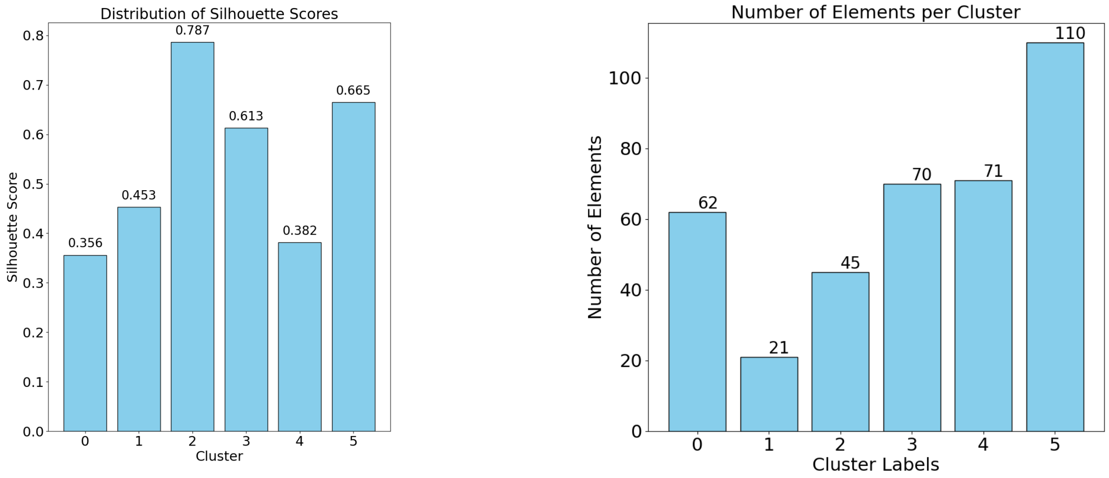
Cluster 26: ERV notifications about issuing the report 'VaR and Peak Position Report By Trader'.

Cluster 27: ERV notifications about issuing the report 'West VaR and Off-Peak Position Report By Trader'.

From these cluster examples, we see that clustering is more granular with these parameters. Considering other parameters could combine all these clusters into a single one since they are similar chains with a template structure.

3. $min\_samples = 30$ $min\_cluster\_size = 20$

   The chains assigned to one of the clusters (excluding noise) have the following length distribution: 356 chains of the length 2, 23 chains of the length 3.



(a) Silhouette score per cluster. **(b)** Number of elements per cluster.

Figure 17. Results for HDBSCAN $min\_samples = 30$ $min\_cluster\_size = 20$.

Cluster 0: Focuses on sales, deals, and checkouts, with potential for automation processes.

Cluster 1: Covers meeting discussions and related topics, without focusing on any specific process.

Cluster 2: Concerns EOL approvals, where every message states 'Please see attached.'

Cluster 3: Pertains to EOL credit responses, where each initial message reads 'The EOL approvals for {date} are attached below.'

Cluster 4: Includes agreements and confirmations, with relatively scattered email chains.

Cluster 5: Related to ERV/TRV notifications. Emails typically say, 'The report named: {report_name} {link}, published as of {date}, is now available for viewing on the website.'

5.3 Summary

After evaluating the clustering results, we have gained several insights and understandings. Although it is challenging to objectively evaluate the processes and select the best parameters for the algorithms due to the lack of a baseline or ground truth, we have still obtained valuable insights.

- With the current pipeline, mainly chains of length 2 and 3 are assigned to clusters, while others tend to be classified as noise. There is potential for future investigation to experiment with different approaches for distance calculations and compare the results with those we obtained.
- The Dunn index and Silhouette score are not completely representative for our use case in process extraction, as they tend to depend on the number of clusters. As the number of clusters increases, clusters can become less distinct and more fragmented, leading to lower values for both metrics.
- We have identified some potential process candidates in our clustering results, which indicates that we have found a way to extract them. However, it is quite difficult to distinguish them from other types of clusters without human evaluation. The current approach also finds already automated and template emails, as well as emails with similar themes such as meeting discussions and congratulations, which are likely candidates for automation.
- Clusters with the highest Silhouette scores usually include already automated emails or templates, as these chains are almost identical. Potential processes of our interest could have lower metric values.
- By varying algorithm parameters, we can achieve different levels of granularity in process extraction. This means that within process groups, some subclusters could be captured by algorithms that result in a greater number of clusters. This information could be helpful in practical situations when selecting parameters.

6 Conclusions

The primary objective of this project was to explore data mining techniques on email data to extract potential business processes that could be automated. This was achieved through a series of well-defined steps, including data preprocessing, email chain detection, text representation using embeddings, and clustering with DBSCAN and HDBSCAN algorithms.

A significant outcome of this study is the identification of clusters that exhibit high degrees of similarity within emails, such as timesheet reminders and various approval processes. These clusters are characterized by repetitive and structured tasks that are ideal candidates for automation.

The evaluation metrics, particularly the Silhouette Score and Dunn Index, provided valuable insights into the quality of our clustering results. Notably, clusters with high Silhouette Scores typically involved templated or already automated emails, underscoring their potential for further larger scale automation.

Our approach demonstrated the flexibility of DBSCAN and HDBSCAN in handling noise and discovering clusters of varying shapes and sizes. By adjusting algorithm parameters, we were able to achieve different levels of granularity in process extraction, which is later crucial for adapting the automation pipeline to specific organizational needs.

Despite the robustness of our clustering algorithms, human evaluation remains essential to distinguish meaningful processes from other types of clusters. This manual assessment helps refine the identified automation candidates, ensuring they are practical and beneficial for the specific case.

In summary, this project highlights the significant potential of leveraging machine learning and data mining techniques to streamline business operations. Automating repetitive email tasks not only enhances productivity but also allows employees to focus on more strategic and innovative activities. The pipeline developed here serves as an example framework that can be adapted for various datasets and domains, facilitating further research and practical applications in process automation. The insights gained from this study pave the way for future exploration into automated solutions.

References

[1] Apoorv Agarwal et al. "A Comprehensive Gold Standard for the Enron Organizational Hierarchy". In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by Haizhou Li et al. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 161–165. URL: `https://aclanthology.org/P12-2032`.

[2] Karl Bringmann et al. *Dynamic Dynamic Time Warping*. 2023. arXiv: `2310.18128 [cs.CG]`.

[3] Jacques Bughin et al. *Skill Shift: Automation and the Future of the Workforce*. Accessed: 2024-05-01. 2018. URL: `https://www.mckinsey.com/featured-insights/future-of-work/skill-shift-automation-and-the-future-of-the-workforce`.

[4] Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. "Density-Based Clustering Based on Hierarchical Density Estimates". In: *Advances in Knowledge Discovery and Data Mining*. Ed. by Jian Pei et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 160–172. ISBN: 978-3-642-37456-2.

[5] Deloitte. *Five 2023 workflow automation trends reshaping the future*. Accessed: 2024-05-01. 2023. URL: `https://www.deloitte.com/global/en/alliances/servicenow/about/2023-workflow-automation-trends.html`.

[6] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[7] Jana Diesner and Kathleen M Carley. "Exploration of communication networks from the Enron email corpus". In: *Proceedings of the 2005 SIAM international conference on data mining*. SIAM. 2005, pp. 3–14.

[8] Jana Diesner, Terrill L. Frantz, and Kathleen M. Carley. "Communication networks from the Enron email corpus "It's always about the people. Enron is no different"". English (US). In: *Computational and Mathematical Organization Theory* 11.3 (Oct. 2005), pp. 201–228. ISSN: 1381-298X. DOI: `10.1007/s10588-005-5377-0`.

[9] J. C. Dunn†. "Well-Separated Clusters and Optimal Fuzzy Partitions". In: *Journal of Cybernetics* 4.1 (1974), pp. 95–104. DOI: `10.1080/01969727408546059`. eprint: `https://doi.org/10.1080/01969727408546059`. URL: `https://doi.org/10.1080/01969727408546059`.

[10] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: KDD'96. Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[11] Absalom E. Ezugwu et al. "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects". In: *Eng. Appl. Artif. Intell.* 110.C (Apr. 2022). ISSN: 0952-1976. DOI: `10.1016/j.engappai.2022.104743`. URL: `https://doi.org/10.1016/j.engappai.2022.104743`.

[12] Hugging Face. *all-mpnet-base-v1*. Accessed: 2024-04-09. 2020. URL: `https://huggingface.co/sentence-transformers/all-mpnet-base-v1`.

[13] Yingying He, Xiaobing Pei, and Lihong Shen. *Semi-supervised learning via DQN for log anomaly detection*. 2024. arXiv: `2401.03151`.

[14] Armand Joulin et al. *Bag of Tricks for Efficient Text Classification*. 2016. arXiv: `1607.01759 [cs.CL]`.

[15] Abhishek Kathuria, Devarshi Mukhopadhyay, and Narina Thakur. “Evaluating Cohesion Score with Email Clustering”. In: *Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*. Ed. by Pradeep Kumar Singh et al. Singapore: Springer Singapore, 2020, pp. 107–119. ISBN: 978-981-15-3369-3.

[16] Bryan Klimt and Yiming Yang. “The Enron Corpus: A New Dataset for Email Classification Research”. In: *Machine Learning: ECML 2004*. Ed. by Jean-François Boulicaut et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 217–226. ISBN: 978-3-540-30115-8.

[17] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. *Numba: A LLVM Compiler for Python Array Expression*. `http://numba.pydata.org/`. Accessed: 2024-04-24. 2015. URL: `http://numba.pydata.org/`.

[18] Xiao Liu, Zhunchen Luo, and Heyan Huang. “Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Ed. by Ellen Riloff et al. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 1247–1256. DOI: `10.18653/v1/D18-1156`. URL: `https://aclanthology.org/D18-1156`.

[19] Adit Magotra. *Actionable Phrase Detection using NLP*. 2022. arXiv: `2210.16841`.

[20] J. Manyika et al. “A future that works: automation, employment, and productivity”. In: 2017. URL: `https://api.semanticscholar.org/CorpusID:114479521`.

[21] Tomas Mikolov et al. “Distributed representations of words and phrases and their compositionality”. In: *Advances in neural information processing systems* (2013), pp. 3111–3119.

[22] Jeffrey Pennington, Richard Socher, and Christopher D Manning. “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[23] CALO Project. *CALO Enron Email Dataset*. `https://enrondata.readthedocs.io/en/latest/data/calo-enron-email-dataset/`. 2015.

[24] Nadia Rahmah and Imas Sukaesih Sitanggang. “Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra”. In: *IOP Conference Series: Earth and Environmental Science* 31.1 (2016), p. 012012. DOI: `10.1088/1755-1315/31/1/012012`.

[25] Nils Reimers and Iryna Gurevych. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Accessed: 2024-04-09. 2019. arXiv: `1908.10084 [cs.CL]`. URL: `https://www.sbert.net/`.

[26] Tim Repke and Ralf Krestel. “Bringing Back Structure to Free Text Email Conversations with Recurrent Neural Networks”. In: *European Conference on Information Retrieval*. 2018. URL: `https://api.semanticscholar.org/CorpusID:4642077`.

[27] Peter J. Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53–65. ISSN: 0377-0427. DOI: `https://doi.org/10.1016/0377-0427(87)90125-7`. URL: `https://www.sciencedirect.com/science/article/pii/0377042787901257`.

[28] Jeanette Samuelsen, Weiqin Chen, and Barbara Wasson. “Integrating multiple data sources for learning analytics—review of literature”. In: *Research and Practice in Technology Enhanced Learning* 14 (2019), p. 11. DOI: `10.1186/s41039-019-0105-4`.

[29] Erich Schubert et al. “DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN”. In: 42.3 (July 2017). ISSN: 0362-5915. DOI: `10.1145/3068335`. URL: `https://doi.org/10.1145/3068335`.

[30] Krupal Shah et al. “Email User Classification and Topic Modeling”. In: *Proceedings of the Future Technologies Conference (FTC) 2020, Volume 1*. Ed. by Kohei Arai, Supriya Kapoor, and Rahul Bhatia. Cham: Springer International Publishing, 2021, pp. 359–377. ISBN: 978-3-030-63128-4.

[31] Jitesh Shetty and Jafar Adibi. “The Enron Email Dataset Database Schema and Brief Statistical Report”. In: 2004. URL: `https://api.semanticscholar.org/CorpusID:59919272`.

[32] Smartbridge. *2023 Automation Trends: Unlock the Future of Intelligent Automation*. Accessed: 2024-05-01. 2023. URL: `https://smartbridge.com/2023-automation-trends-unlock-the-future-of-intelligent-automation/`.

[33] Kaitao Song et al. “MPNet: Masked and Permuted Pre-training for Language Understanding”. In: *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*. 2020. URL: `https://arxiv.org/abs/2004.09297`.

[34] Congcong Wang, Paul Nulty, and David Lillis. “A Comparative Study on Word Embeddings in Deep Learning for Text Classification”. In: *Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval*. NLPIR ’20. Seoul, Republic of Korea: Association for Computing Machinery, 2021, pp. 37–46. ISBN: 9781450377607. DOI: `10.1145/3443279.3443304`. URL: `https://doi.org/10.1145/3443279.3443304`.

[35] Xia Wang et al. “Email Conversations Reconstruction Based on Messages Threading for Multi-person”. In: *2008 International Workshop on Education Technology and Training and 2008 International Workshop on Geoscience and Remote Sensing*. Vol. 1. 2008, pp. 676–680. DOI: `10.1109/ETTandGRS.2008.321`.

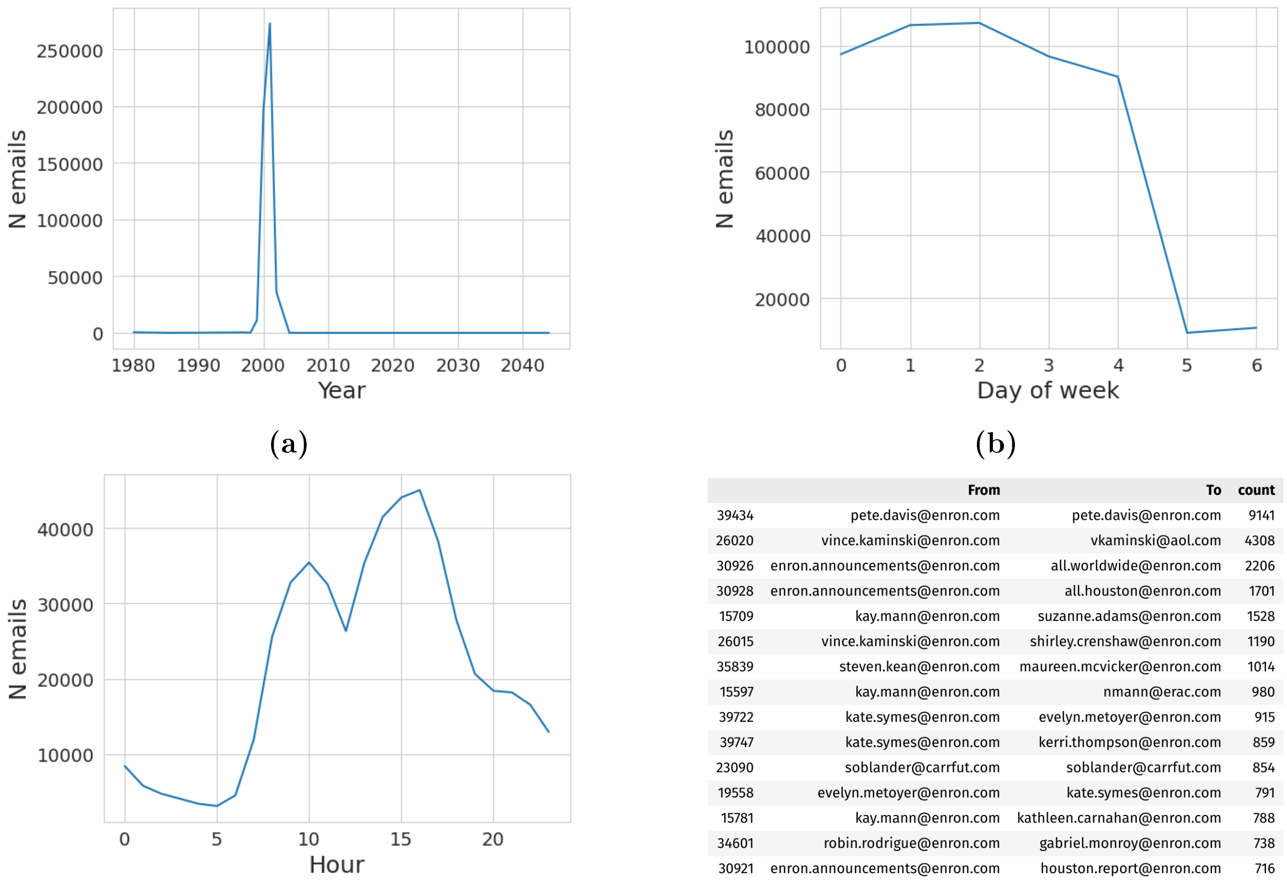
[36] Jen-Yuan Yeh. “Email Thread Reassembly Using Similarity Matching”. In: *International Conference on Email and Anti-Spam*. 2006. URL: `https://api.semanticscholar.org/CorpusID:6206273`.

[37] Linan Yue et al. “Event Grounded Criminal Court View Generation with Cooperative (Large) Language Models”. In: *arXiv preprint arXiv:2404.07001* (2024). URL: `https://arxiv.org/abs/2404.07001`.

7 Appendices

.1 Enron Dataset Information

In Figure 18, information about the distribution of emails is presented. We can observe that most correspondence took place during 2001-2002, primarily on working days and within working hours. In the last image, it is noticeable that a substantial number of emails were sent from and to addresses related not to a single person but to a group. This makes the detection of chains more complicated.



(a) (b) (c) (d)

	From	To	count
39434	pete.davis@enron.com	pete.davis@enron.com	9141
26020	vince.kaminski@enron.com	vkaminski@aol.com	4308
30926	enron.announcements@enron.com	all.worldwide@enron.com	2206
30928	enron.announcements@enron.com	all.houston@enron.com	1701
15709	kay.mann@enron.com	suzanne.adams@enron.com	1528
26015	vince.kaminski@enron.com	shirley.crenshaw@enron.com	1190
35839	steven.kean@enron.com	maureen.mcvicker@enron.com	1014
15597	kay.mann@enron.com	nmann@erac.com	980
39722	kate.symes@enron.com	evelyn.metoyer@enron.com	915
39747	kate.symes@enron.com	kerri.thompson@enron.com	859
23090	soblander@carrfut.com	soblander@carrfut.com	854
19558	evelyn.metoyer@enron.com	kate.symes@enron.com	791
15781	kay.mann@enron.com	kathleen.carnahan@enron.com	788
34601	robin.rodrigue@enron.com	gabriel.monroy@enron.com	738
30921	enron.announcements@enron.com	houston.report@enron.com	716

Figure 18. Distribution of the number of emails based on (a) year, (b) day of the week, and (c) time of day; (d) popular sender-recipient pairs.

.2 Issues in Restoring Chains

In this section, we present examples of issues in the original data that impacted the chain detection process, making it necessary for us to be more meticulous in this step.

```
Message-ID:                <20500759.1075845350975.JavaMail.evans@thyme>
Date:                      Wed, 2 May 2001 19:05:00 -0700 (PDT)
From:                      maria.tefel@enron.com
To:                        v.weldon@enron.com, eric.boyt@enron.com
Subject:                   UNOCAL - Gas pricing Info
Cc:                        john.kiani@enron.com
Mime-Version:              1.0
Content-Type:              text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Bcc:                       john.kiani@enron.com
Eric and Charlie:
Thanks for your time in meeting with us.  Attached is the information on
Unocal.  The minimum required deliverable volumes are based on 40% load
factor, and the put is based on 10% above that (volumes between a 40%
and a 44% load factor).  Thanks and please call me if you have any
questions.
Maria
```

```
Message-ID:                <19206155.1075851684797.JavaMail.evans@thyme>
Date:                      Wed, 2 May 2001 09:05:00 -0700 (PDT)
From:                      maria.tefel@enron.com
To:                        v.weldon@enron.com, eric.boyt@enron.com
Subject:                   UNOCAL - Gas pricing Info
Cc:                        john.kiani@enron.com
Mime-Version:              1.0
Content-Type:              text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Bcc:                       john.kiani@enron.com
Eric and Charlie:
Thanks for your time in meeting with us.  Attached is the information on
Unocal.  The minimum required deliverable volumes are based on 40% load
factor, and the put is based on 10% above that (volumes between a 40%
and a 44% load factor).  Thanks and please call me if you have any
questions.
Maria
```

Figure 19. Example of time error in the duplicate emails (Computer-generated metadata is omitted)

```
Message-ID:               16486172.1075842795064.JavaMail.evans@thyme
Date:                     Wed, 16 May 2001 01:33:00 -0700 (PDT)
From:                     theresa.staab@enron.com
To:                       debra.perlingiere@enron.com,
                          gerald.nemec@enron.com
Subject:                  May Confirmation Information
Mime-Version:             1.0
Content-Type:             text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
May Confirmation Information:
Kennedy:  13,855 MMBtu, IF CIG Index minus $.67
North Finn:  409 MMBtu, IF CIG Index minus $.67
Quantum:  557 MMBtu, IF CIG Index minus $.67
Wellstar (May through July):  1232 MMBtu, IF CIG Index minus $.67
Let me know if you need more information,
Thanks, Theresa
(303) 575-6485
```

```
Message-ID:               <24159726.1075842795088.JavaMail.evans@thyme>
Date:                     Wed, 16 May 2001 02:01:00 -0700 (PDT)
From:                     theresa.staab@enron.com
To:                       debra.perlingiere@enron.com,
                          gerald.nemec@enron.com
Subject:                  May Confirmation Information
Mime-Version:             1.0
Content-Type:             text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
Wellstar is minus $.70 (since it was for 3 months).
Theresa
(303) 575-6485
--- Forwarded by Theresa Staab/Corp/Enron on 05/16/2001 09:00 AM ---
Theresa Staab
05/16/2001 08:33 AM
To:  Debra Perlingiere/HOU/ECT@ECT, Gerald Nemec/HOU/ECT@ECT
cc:
Subject:  May Confirmation Information
May Confirmation Information:
Kennedy:  13,855 MMBtu, IF CIG Index minus $.67
North Finn:  409 MMBtu, IF CIG Index minus $.67
Quantum:  557 MMBtu, IF CIG Index minus $.67
Wellstar (May through July):  1232 MMBtu, IF CIG Index minus $.67
Let me know if you need more information,
Thanks, Theresa
(303) 575-6485
```

Figure 20. Example of a missing "Fwd" (Computer-generated metadata is omitted)

```
Message-ID:                 <31191677.1075847034776.JavaMail.evans@thyme>
Date:                       Tue, 31 Oct 2000 01:34:00 -0800 (PST)
From:                       bob.shults@enron.com
To:                         michael.slade@enron.com
Subject:                    GFI Non disclosure
Cc:                         tana.jones@enron.com
Mime-Version:               1.0
Content-Type:               text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit
Bcc:                        tana.jones@enron.com
I am working with Andy Zipper on the Broker Client application.  I talked to
Colin Hefron at GFI and his attorney is apparently working with you on the
Non Disclosure Agreement.  Please let me know the status so that I can move
forward on discussions with GFI. My number is 713 853-0397.
```

```
Message-ID:                 <19646752.1075847034849.JavaMail.evans@thyme>
Date:                       Tue, 31 Oct 2000 02:09:00 -0800 (PST)
From:                       tana.jones@enron.com
To:                         bob.shults@enron.com
Subject:                    Re:  GFI Non disclosure
Mime-Version:               1.0
Content-Type:               text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit
When we can get a copy of this, we need to make sure it's broad enough to
cover what you're doing...
--- Forwarded by Tana Jones/HOU/ECT on 10/31/2000 10:09 AM ---
Michael Slade
10/31/2000 09:49 AM
To:  Bob Shults/HOU/ECT@ECT
cc:  Tana Jones/HOU/ECT@ECT
Subject:  Re:  GFI Non disclosure
Hello Bob
Yes that's right - although it should all be done now, although I had a call
saying they hadn't received the doc, although I did courier it round.
Unfortunately I am having to work from home today because of flooding in
south of England - so I can't check the file - but will do so as soon as I
get back to the office
Michael
To:  michael.slade@enron.com
cc:  Tana Jones/HOU/ECT@ECT
Subject:  GFI Non disclosure
I am working with Andy Zipper on the Broker Client application.  I talked to
Colin Hefron at GFI and his attorney is apparently working with you on the
Non Disclosure Agreement.  Please let me know the status so that I can move
forward on discussions with GFI. My number is 713 853-0397.
```

Figure 21. Example of an intermidiate email missing (Computer-generated metadata is omitted)

```
Message-ID:               <11500543.1075841633985.JavaMail.evans@thyme>
Date:                     Wed, 3 Jan 2001 07:25:00 -0800 (PST)
From:                     kate.symes@enron.com
To:                       sharen.cason@enron.com
Subject:                  Re:  #488882
Mime-Version:             1.0
Content-Type:             text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
I've changed the deal to be confirmed.  Mike Driscoll changed this deal
earlier, which must have affected the confirmation status.  Thanks.
Kate
From:  Sharen Cason 01/03/2001 03:18 PM
To:  Kate Symes/PDX/ECT@ECT
cc:  Kimberly Hundl/Corp/Enron@Enron
Subject:  #488882
This deal is entered as not to be confirmed.  It looks like it should be
confirmed.  Can you check into this and let me know.
Thanks!
```

```
Message-ID:               <21340799.1075841633962.JavaMail.evans@thyme>
Date:                     Wed, 3 Jan 2001 09:18:00 -0800 (PST)
From:                     sharen.cason@enron.com
To:                       kate.symes@enron.com
Subject:                  #488882
Cc:                       kimberly.hundl@enron.com
Mime-Version:             1.0
Content-Type:             text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
Bcc:                      kimberly.hundl@enron.com
This deal is entered as not to be confirmed.  It looks like it should be
confirmed.  Can you check into this and let me know.
Thanks!
```

Figure 22. Example of time error in chains (Computer-generated metadata is omitted)

```
Message-ID:                  <6656940.1075856256946.JavaMail.evans@thyme>
Date:                        Wed, 22 Nov 2000 02:01:00 -0800 (PST)
From:                        vince.kaminski@enron.com
To:                          tconvery@wharton.upenn.edu
Subject:                     Re:  December 6th Meeting
Cc:                          vince.kaminski@enron.com
Mime-Version:                1.0
Content-Type:                text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
Bcc:                         vince.kaminski@enron.com
Theresa,
Thanks.  I appreciate it.
Happy Thanksgiving and please give my regards and best wishes to Howard.
Vince
"Convery, Theresa" <tconvery@wharton.upenn.edu> on 11/22/2000 09:39:53 AM
To:  "Vince Kaminski (E-mail)" <vkamins@enron.com>
cc:  "Kunreuther, Howard" <kunreuth@wharton.upenn.edu>
Subject:  December 6th Meeting
Dear Mr.  Kaminski:
This is to confirm the December 6th Meeting here at our Center.
The location for the meeting is Room # 3212 Steinberg Hall-Dietrich Hall
and the time will run from 9:00 AM - 11:00 AM. Please let us know if you
need anything further.  We look forward to seeing you then.
Regards,
Theresa Convery
(...)
```

```
Message-ID:                  <33372255.1075856257153.JavaMail.evans@thyme>
Date:                        Wed, 22 Nov 2000 02:39:00 -0800 (PST)
From:                        tconvery@wharton.upenn.edu
To:                          vkamins@enron.com
Subject:                     December 6th Meeting
Cc:                          kunreuth@wharton.upenn.edu
Mime-Version:                1.0
Content-Type:                text/plain; charset=us-ascii
Content-Transfer-Encoding:7bit
Bcc:                         kunreuth@wharton.upenn.edu
Dear Mr.  Kaminski:
This is to confirm the December 6th Meeting here at our Center.
The location for the meeting is Room # 3212 Steinberg Hall-Dietrich Hall
and the time will run from 9:00 AM - 11:00 AM. Please let us know if you
need anything further.  We look forward to seeing you then.
Regards,
Theresa Convery
(...)
```

Figure 23. Example of one person owning several email addresses (Computer-generated metadata is omitted)

.3 Clusters Examples

Here, we present portions of representative clusters to give an impression of what the clustering results look like.

HDBSCAN $min\_samples = 10$, $min\_cluster\_size = 20$

Cluster 1: meeting emails with dates and times.

Chain 1/51:

```
Email 1:

Message-ID: <9616314.1075839936470.JavaMail.evans@thyme>
Date:  Fri, 27 Apr 2001 22:59:00 -0700 (PDT)
From:  dan.dietrich@enron.com
To:  murray.o'neil@enron.com, david.steiner@enron.com, bill.iii@enron.com
Subject:  Meeting
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Gentlemen,
I would like to have a conference call to discuss recent IT issues.
How does 3:45pm CST (1:45pm PST) sound...please advise..
Dan


Email 2:

Message-ID: <19116636.1075839936442.JavaMail.evans@thyme>
Date:  Fri, 27 Apr 2001 23:24:00 -0700 (PDT)
From:  dan.dietrich@enron.com
To:  david.steiner@enron.com, murray.o'neil@enron.com, bill.iii@enron.
Subject:  RE: Meeting
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Sounds good.

---Original Message---
From:  Steiner, David
Sent:  Friday, April 27, 2001 3:10 PM
To:  Dietrich, Dan; O'Neil, Murray; Williams III, Bill
Subject:  RE: Meeting
I am good with that, Bill may not be available as I think he
is out of the office at a class.
Do you want us to call you at your desk?
Dave
```

```
---Original Message---
From:  Dietrich, Dan
Sent:  Friday, April 27, 2001 12:59 PM
To:  O'Neil, Murray; Steiner, David; Williams III, Bill
Subject:  Meeting
Gentlemen,
I would like to have a conference call to discuss recent IT issues.
How does 3:45pm CST (1:45pm PST) sound...please advise..
Dan
```

Chain 2/51:

```
Email 1:

Message-ID: <722569.1075860900777.JavaMail.evans@thyme>
Date:  Fri, 25 Jan 2002 15:29:58 -0800 (PST)
From:  eric.gadd@enron.com
To:  susan.wadle@enron.
Subject:  Meeting
Cc:  stephen.dowd@enron.com, kimberly.watson@enron.com,
tracy.geaccone@enron.com
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit
Bcc:  stephen.dowd@enron.com, kimberly.watson@enron.com,
tracy.geaccone@enron.com

Susan,
please organize a meeting with Steve, Kim, and Tracey early next
week, say Monday or Tuesday
Agenda-
Project depreciation
ROE analysis
Financial accounting vs regulatory accounting

Email 2:

Message-ID: <15607773.1075860921400.JavaMail.evans@thyme>
Date:  Mon, 28 Jan 2002 06:46:50 -0800 (PST)
From:  susan.wadle@enron.com
To:  kimberly.watson@enron.com
Subject:  RE: Meeting
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

As soon as we set the time I will invite them.

---Original Message---
```

```
From:  Watson, Kimberly
Sent:  Monday, January 28, 2002 8:45 AM
To:  Wadle, Susan
Subject:  FW: Meeting
Susan,
Also, please include James Centilli and Mark McConnell.
Thanks, Kim.

---Original Message---
From:  Gadd, Eric
Sent:  Friday, January 25, 2002 5:30 PM
To:  Wadle, Susan
Cc:  Dowd, Stephen; Watson, Kimberly; Geaccone, Tracy
Subject:  Meeting

Susan,
please organize a meeting with Steve, Kim, and Tracey early next
week, say Monday or Tuesday
Agenda-
Project depreciation
ROE analysis
Financial accounting vs regulatory accounting
```

Chain 3/51:

```
Email 1:

Message-ID: <26068317.1075861080705.JavaMail.evans@thyme>
Date:  Mon, 25 Feb 2002 08:53:07 -0800 (PST)
From:  shelley.corman@enron.com
To:  ricki.winters@enron.com
Subject:  Schedule
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Ricki - would you please:
1.  Make reservations at 11:45 on Thurs for Judy Johnson & I at Duck & Co
in Hyatt
2.  Set up a pre-meeting for Don Hawkins & I before our meeting with Rod
tomorrow
3.  Move my Wed.  staff meeting to run from 10-11:30 (instead of 9-10).
Since we no longer have a Wash office, I plan to watch the Commission
meeting myself

Email 2:

Message-ID: <27967603.1075861093391.JavaMail.evans@thyme>
```

```
Date:  Mon, 25 Feb 2002 09:30:56 -0800 (PST)
From:  ricki.winters@enron.com
To:  shelley.corman@enron.com
Subject:  RE: Schedule
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Reservations made at Ducks on Thursday, meeting with Don is at 2:30 today
in your office and staff meeting has been moved to requested time.
Thank you, Ricki

---Original Message---
From:  Corman, Shelley
Sent:  Monday, February 25, 2002 10:53 AM
To:  Winters, Ricki
Subject:  Schedule
Ricki - would you please:
1.  Make reservations at 11:45 on Thurs for Judy Johnson & I at Duck & Co
in Hyatt
2.  Set up a pre-meeting for Don Hawkins & I before our meeting with Rod
tomorrow
3.  Move my Wed.  staff meeting to run from 10-11:30 (instead of 9-10).
Since we no longer have a Wash office, I plan to watch the Commission
meeting myself
```

Chain 4/51:

```
Email 1:

Message-ID: <32287685.1075858793813.JavaMail.evans@thyme>
Date:  Tue, 16 Oct 2001 08:49:59 -0700 (PDT)
From:  w..white@enron.com
To:  john.postlethwaite@enron.com, martha.stevens@enron.com,
casey.evans@enron.com
Subject:  today's meeting
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

We will be having a staff meeting today.  If we can start at 12:15 rather
than 12:00 that would be great because I have an 11:00 meeting.
John,
What is the phone number where we can reach you?
Sorry I do not have an itinerary.  I would like to go over Doorstep
findings and anything else you would like to fill me in on since we have
not met in a while.
Stacey
```

```
Email 2:

Message-ID: <18506233.1075858769325.JavaMail.evans@thyme>
Date:  Tue, 16 Oct 2001 08:55:37 -0700 (PDT)
From:  john.postlethwaite@enron.com
To:  w..white@enron.com
Subject:  RE: today’s meeting
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

The number is 503-464-7583.  I will be at this number for all our staff
meetings.
John

---Original Message---
From:  White, Stacey W.
Sent:  Tuesday, October 16, 2001 8:50 AM
To:  Postlethwaite, John; Stevens, Martha; Evans, Casey
Subject:  today’s meeting
We will be having a staff meeting today.  If we can start at 12:15 rather
than 12:00 that would be great because I have an 11:00 meeting.
John,
What is the phone number where we can reach you?
Sorry I do not have an itinerary.  I would like to go over Doorstep
findings and anything else you would like to fill me in on since we have
not met in a while.
Stacey
```

Cluster 0: checkouts, invoices, bookings.

Chain 1/126:

```
Email 1:

Message-ID: <28606955.1075841663057.JavaMail.evans@thyme>
Date:  Mon, 26 Feb 2001 07:18:00 -0800 (PST)
From:  kate.symes@enron.com
To:  kerri.thompson@enron.com
Subject:  Re:  apb
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Deal had no broker - 531846 has been changed to reflect APB as broker.
Thanks,
Kate
```

```
Kerri Thompson@ENRON
02/26/2001 02:35 PM
To:  Kate Symes/PDX/ECT@ECT
cc:
Subject:  apb
missing deal
trader?
sell mirant
28th
25 mw
195.00
mid c

Email 2:

Message-ID: <10042269.1075841662751.JavaMail.evans@thyme>
Date:  Mon, 26 Feb 2001 08:35:00 -0800 (PST)
From:  kerri.thompson@enron.com
To:  kate.symes@enron.com
Subject:  apb
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

missing deal
trader?
sell mirant
28th
25 mw
195.00
mid c
```

Chain 2/126:

```
Email 1:

Message-ID: <12954685.1075841674671.JavaMail.evans@thyme>
Date:  Tue, 13 Mar 2001 06:40:00 -0800 (PST)
From:  kate.symes@enron.com
To:  kerri.thompson@enron.com
Subject:  Re:  apb checkout
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

chris - just finished entering deals
jeff - 549013 - changed to APB from Prebon
548756 - changed to 82.52
```

```
Kerri Thompson@ENRON
03/13/2001 02:28 PM
To:  Kate Symes/PDX/ECT@ECT
cc:
Subject:  apb checkout
missing deals

chris
selling bp energy
111.00
25 mw
sp15
april
off peak

jeff
selling bp energy
205.00
25 mw
pv
april
on peak

548756
broker has 82.52

thanks

Email 2:

Message-ID: <25112024.1075841674553.JavaMail.evans@thyme>
Date:  Tue, 13 Mar 2001 08:28:00 -0800 (PST)
From:  kerri.thompson@enron.com
To:  kate.symes@enron.com
Subject:  apb checkout
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

missing deals

chris
selling bp energy
111.00
25 mw
sp15
april
off peak
```

```
jeff
selling bp energy
205.00
25 mw
pv
april
on peak

548756
broker has 82.52

thanks
```

Chain 3/126:

```
Email 1:

Message-ID: <18050985.1075841687965.JavaMail.evans@thyme>
Date:  Fri, 30 Mar 2001 03:02:00 -0800 (PST)
From:  kate.symes@enron.com
To:  stephanie.piwetz@enron.com
Subject:  Re:  Deal 562195.01
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

it's been changed.

Stephanie Piwetz 03/30/2001 11:07 AM
To:  Kate Symes/PDX/ECT@ECT
cc:
Subject:  Re:  Deal 562195.01
Kate, can you please correct the fee for del 562195.01, to .0275

THanks
Sp

Email 2:

Message-ID: <6082278.1075841687942.JavaMail.evans@thyme>
Date:  Fri, 30 Mar 2001 05:07:00 -0800 (PST)
From:  stephanie.piwetz@enron.com
To:  kate.symes@enron.com
Subject:  Re:  Deal 562195.01
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit
```

```
Kate, can you please correct the fee for del 562195.01, to .0275

THanks
Sp
```

Cluster 2: congratulations, mainly on promotions.

Chain 1/52:

```
Email 1:

Message-ID: <29043371.1075851705096.JavaMail.evans@thyme>
Date:  Tue, 30 Jan 2001 23:27:00 -0800 (PST)
From:  jean.bell@enron.com
To:  errol.mclaughlin@enron.com
Subject:  CONGRATULATIONS !
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

I just learned of your promotion to Senior Specialist, and I wanted to
say "Congratulations"; it could not have happened to a better person !

Jean

Email 2:

Message-ID: <26146173.1075857380851.JavaMail.evans@thyme>
Date:  Wed, 31 Jan 2001 03:25:00 -0800 (PST)
From:  errol.mclaughlin@enron.com
To:  jean.bell@enron.com
Subject:  Re:  CONGRATULATIONS !
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Thank you very much Jean.  I apreciate it.
```

Chain 2/52:

```
Email 1:

Message-ID: <13843856.1075856778553.JavaMail.evans@thyme>
Date:  Wed, 19 Jan 2000 08:46:00 -0800 (PST)
From:  kimberly.watson@enron.com
To:  vince.kaminski@enron.com
Subject:  promotion
Mime-Version:  1.0
```

```
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Vince, I want to congratulate you on your promotion to Managing Director!
As I scanned the list of people who were promoted, I was so pleased to
see your name on the list.  As large as Enron is, it is refreshing to
see people like you with incredible skill and talent receive deserving
promotions.  I have certainly enjoyed working with you and the R&D team
over the past year and look forward to a successful 2000 as we break new
ground for ET&S. Kim.

Email 2:

Message-ID: <16124015.1075856778478.JavaMail.evans@thyme>
Date:  Thu, 20 Jan 2000 00:11:00 -0800 (PST)
From:  vince.kaminski@enron.com
To:  kimberly.watson@enron.com
Subject:  Re:  promotion
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Kim,
Thanks a lot.  I appreciate your kind words.
Vince

Kimberly Watson@ENRON
01/19/2000 04:46 PM
To:  Vince J Kaminski/HOU/ECT@ECT
cc:
Subject:  promotion
Vince, I want to congratulate you on your promotion to Managing Director!
As I scanned the list of people who were promoted, I was so pleased to
see your name on the list.  As large as Enron is, it is refreshing to
see people like you with incredible skill and talent receive deserving
promotions.  I have certainly enjoyed working with you and the R&D team
over the past year and look forward to a successful 2000 as we break new
ground for ET&S. Kim.
```

Chain 3/52:

```
Email 1:

Message-ID: <32154438.1075849815673.JavaMail.evans@thyme>
Date:  Tue, 16 Jan 2001 00:38:00 -0800 (PST)
From:  thomas.gros@enron.com
To:  sally.beck@enron.com
Subject:  Congratulations!
```

```
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Congratulations on your promotion to MD!
In addition to being a great personal achievement, your promotion helps
to raise the awareness of the importance of the tasks performed by you
and your team.

I look forward to speaking with you upon your return from Europe.

All the best,
Tom

Email 2:

Message-ID: <32002433.1075855952483.JavaMail.evans@thyme>
Date:  Tue, 16 Jan 2001 00:46:00 -0800 (PST)
From:  sally.beck@enron.com
To:  thomas.gros@enron.com
Subject:  Re:  Congratulations!
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

Thanks for your nice note.  I do hope that everyone on my team sees the
promotion as recognition of the value of the roles that we perform.  All
that should make it easier to attract and retain talented people that
make my job easier and more fun!

I will ask Patti to schedule some time for us to get together when I am
back in the office next Monday.  Have a good week.  --Sally

Thomas D Gros@ENRON
01/16/2001 08:38 AM
To:  Sally Beck/HOU/ECT@ECT
cc:
Subject:  Congratulations!
Congratulations on your promotion to MD!
In addition to being a great personal achievement, your promotion helps
to raise the awareness of the importance of the tasks performed by you
and your team.

I look forward to speaking with you upon your return from Europe.

All the best,
Tom
```

Cluster 10: ERV Notification automated emails.

Chain 1/110:

```
Email 1:

Message-ID: <5844272.1075861374003.JavaMail.evans@thyme>
Date:  Tue, 20 Nov 2001 05:42:54 -0800 (PST)
From:  john.allison@enron.com
To:  chris.abel@enron.com, john.allison@enron.com,
c..gossett@enron.com, frank.hayden@enron.com, louise.kitchen@enron.com,
john.lavorato@enron.com, david.patton@enron.com, david.port@enron.com,
kenneth.thibodeaux@enron.com, tom.victorio@enron.com,
cassi.wallace@enron.com, greg.whalley@enron.com, w..white@enron.com,
shona.wilson@enron.com
Subject:  ERV Notification:  (Enron Americas Position Report -
11/19/2001)
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

The report named:  Enron Americas Position Report {link}, published as of
11/19/2001 is now available for viewing on the website.

Publisher's Notes:
PRELIM - The risktRAC VaR calculation is still running.

Email 2:

Message-ID: <5261504.1075861374409.JavaMail.evans@thyme>
Date:  Tue, 20 Nov 2001 13:09:43 -0800 (PST)
From:  david.patton@enron.com
To:  chris.abel@enron.com, john.allison@enron.com,
c..gossett@enron.com, frank.hayden@enron.com, louise.kitchen@enron.com,
john.lavorato@enron.com, david.patton@enron.com, david.port@enron.com,
kenneth.thibodeaux@enron.com, tom.victorio@enron.com,
cassi.wallace@enron.com, greg.whalley@enron.com, w..white@enron.com,
shona.wilson@enron.com
Subject:  ERV Notification:  (Enron Americas Position Report -
11/19/2001)
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

The report named:  Enron Americas Position Report {link}, published as of
11/19/2001 is now available for viewing on the website.

Publisher's Notes:
FINAL
```

Chain 2/110:

```
Email 1:

Message-ID: <15587025.1075862155749.JavaMail.evans@thyme>
Date:  Tue, 30 Oct 2001 18:06:47 -0800 (PST)
From:  melissa.videtto@enron.com
To:  chuck.ames@enron.com, f..brawner@enron.com, alejandra.chavez@enron.com,
darren.espey@enron.com, kulvinder.fowler@enron.com,
chris.germany@enron.com, scott.goodell@enron.com, john.hodge@enron.com,
james.hungerford@enron.com, luchas.johnson@enron.com,
f..keavey@enron.com, kam.keiser@enron.com, brad.mckay@enron.com,
jonathan.mckay@enron.com, hal.mckinney@enron.com,
scott.neal@enron.com, scott.palmer@enron.com, w..pereira@enron.com,
vladi.pimenov@enron.com, andrea.ring@enron.com, jeff.royed@enron.com,
kimat.singla@enron.com, craig.taylor@enron.com, judy.townsend@enron.com,
victoria.versen@enron.com, melissa.videtto@enron.com,
ashley.worthing@enron.com
Subject:  TRV Notification:  (East P/L Totals - 10/30/2001)
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-
Content-Transfer-Encoding:  7bit

The report named:  East P/L Totals {link}, published as of 10/30/2001 is
now available for viewing on the website.

Email 2:

Message-ID: <8212720.1075862154795.JavaMail.evans@thyme>
Date:  Wed, 31 Oct 2001 05:39:40 -0800 (PST)
From:  kam.keiser@enron.com
To:  chuck.ames@enron.com, f..brawner@enron.com, alejandra.chavez@enron.com,
darren.espey@enron.com, kulvinder.fowler@enron.com,
chris.germany@enron.com, scott.goodell@enron.com, john.hodge@enron.com,
james.hungerford@enron.com, luchas.johnson@enron.com,
f..keavey@enron.com, kam.keiser@enron.com, brad.mckay@enron.com,
jonathan.mckay@enron.com, hal.mckinney@enron.com,
scott.neal@enron.com, scott.palmer@enron.com, w..pereira@enron.com,
vladi.pimenov@enron.com, andrea.ring@enron.com, jeff.royed@enron.com,
kimat.singla@enron.com, craig.taylor@enron.com, judy.townsend@enron.com,
victoria.versen@enron.com, melissa.videtto@enron.com,
ashley.worthing@enron.com
Subject:  TRV Notification:  (East P/L Totals - 10/30/2001)
Mime-Version:  1.0
Content-Type:  text/plain; charset=us-ascii
Content-Transfer-Encoding:  7bit

The report named:  East P/L Totals {link}, published as of 10/30/2001 is
now available for viewing on the website.

(Revision:  2)
```

Publisher’s Notes:
Correction: FT-New York and VNG