Manifesto Text Clustering

Table of contents

1 Data

This project utilizes data from the Comparative Manifesto Project (CMP), which provides political manifestos annotated at the quasi-sentence level. These annotations include manually assigned policy codes corresponding to predefined political topics. The objective of this project is to use the text content of these quasi-sentences—without labels—to cluster them into coherent topic groups resembling the CMP's main categories.

1.1 Data Source and Access

The data is accessed programmatically via the CMP's official REST API. An API key is used for authenticated requests. Using a series of endpoint queries, the following resources were retrieved:

- **Core dataset metadata** (party, country, date, etc.)
- **Manifesto metadata** to filter for English translations and machine-readable annotations
- **Full quasi-sentence text** and their respective structure for selected manifestos

Data retrieval is restricted to manifestos from Germany, Switzerland, and Austria, where English translations and machine-annotated versions are available.

1.2 Manifesto Selection Criteria

From the full list of manifestos, only a subset was selected based on the following conditions:

1. Country must be Germany, Switzerland, or Austria
2. Manifesto must have an English translation (`translation_en = True`)
3. Manifesto must include sentence-level annotations (`annotations = True`)
4. Only the latest manifesto per party is kept to avoid duplication and ensure contemporary relevance

```
Daten aus Cache geladen.
```

1.3 Quasi-Sentence Extraction

The text content of each manifesto is returned as a list of quasi-sentences—units of meaning used by CMP to annotate policy content. These were extracted and aggregated into a dictionary of the form:

```
{
    "manifesto_id_1": [
        {"text": "qs1", "cmp_code": 101},
        {"text": "qs2", "cmp_code": 204},
        {"text": "qs3", "cmp_code": 503},
        # ...
    ],
    "manifesto_id_2": [
        {"text": "qs1", "cmp_code": 402},
        {"text": "qs2", "cmp_code": 302},
        # ...
    ],
    # ...
}
```

1.4 Category Mapping

lthough this is an unsupervised task, the CMP codebook was downloaded and consulted to define reference categories for qualitative evaluation. The seven CMP "main domains" used as reference topics are:

1. External Relations
2. Freedom and Democracy

3. Political System
4. Economy
5. Welfare and Quality of Life
6. Fabric of Society
7. Social Groups

These categories serve as a qualitative benchmark to interpret the discovered clusters. While these are not used directly in training or clustering, several hand-coded samples are referenced for evaluation in Section 4.

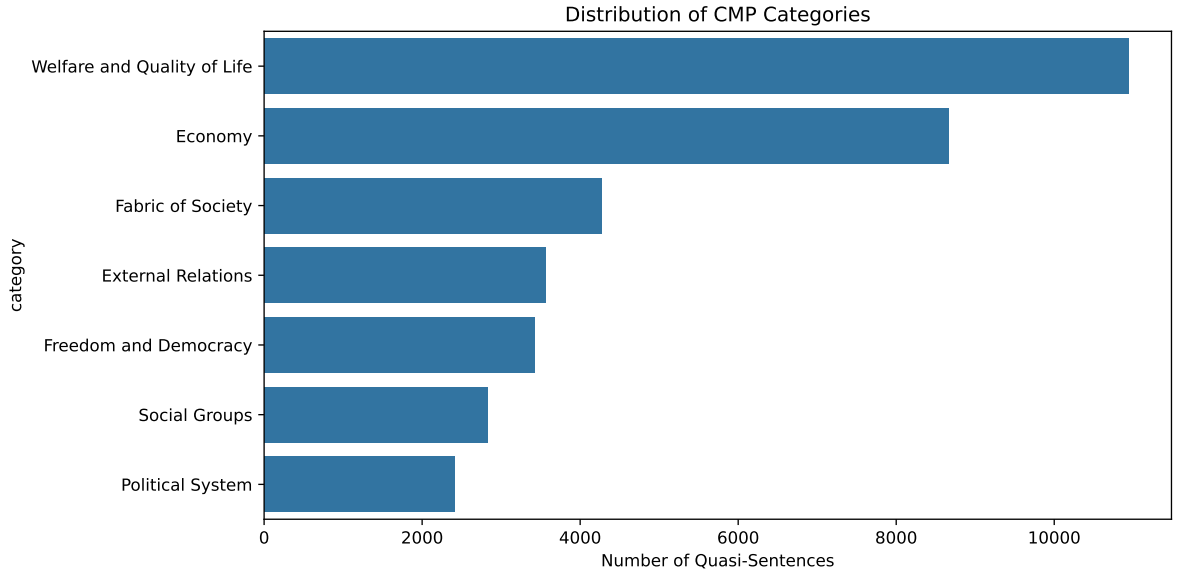
The resulting dataframe looks like this:

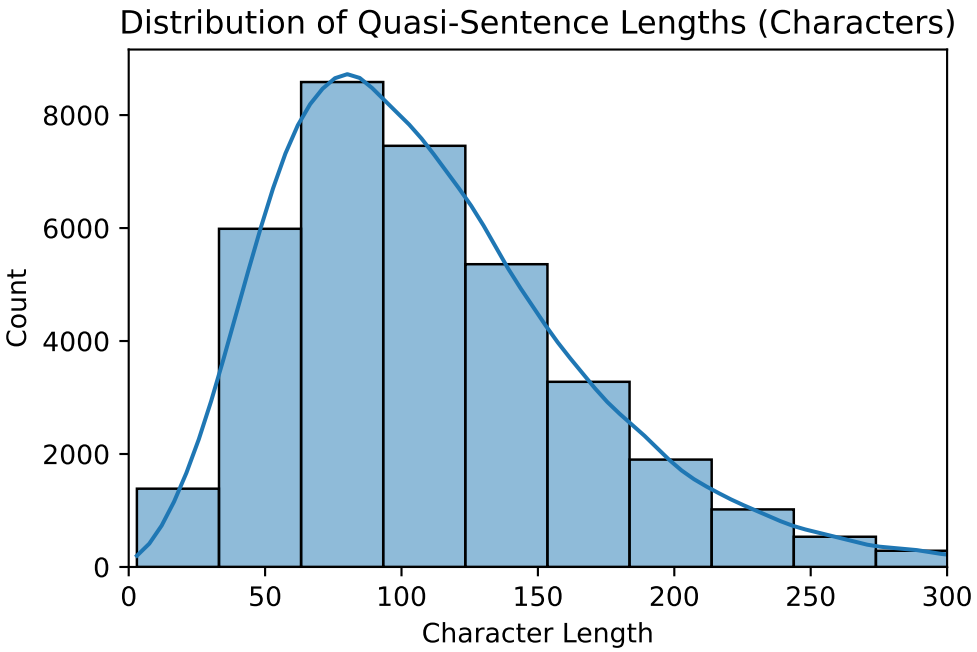
manifesto_id	text	cmp_code	category
GER_202109	"We support..."	403	Economy
GER_202109	"Democracy is essential"	201	Freedom and Democracy

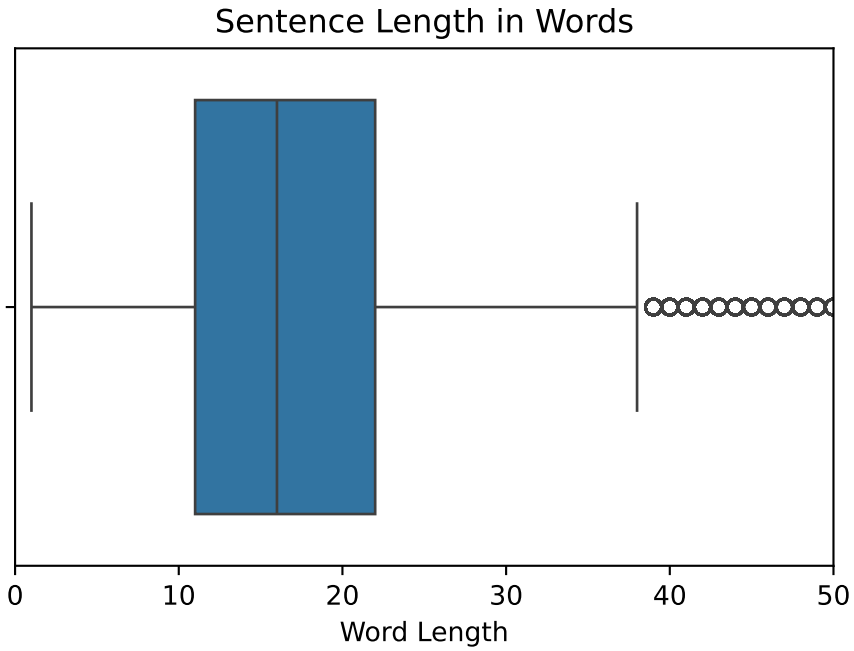
1.5 Data Exploration

First, I explore the Data in the form described previously to get an idea if it is ready for clustering or if it needs pore preprocessing.

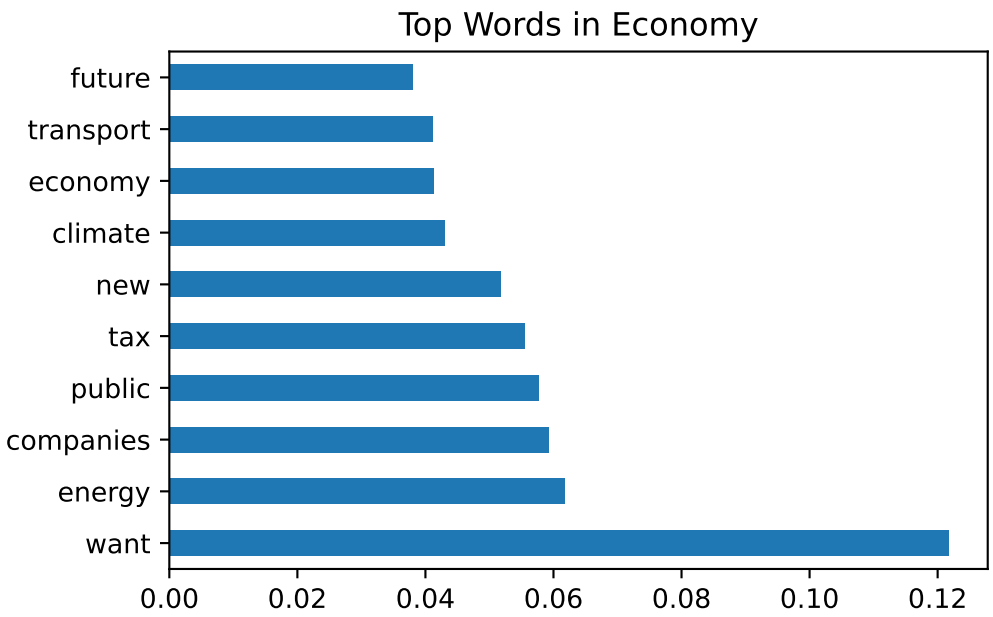
```
Number of quasi-sentences: 36088
Number of unique manifestos: 34
Number of unique categories: 7
```







To get an idea about the content of the categories, I take a look at the most frequent words. Here I take the example of Economy.



1.6 Grouping Sentences

In the Data Exploration Step I realized that the length of the Quasi-Sentences would be a problem. The average length is only about fifteen words which makes the Clustering Task very difficult until almost impossible. For this reason the Quasi-Sentences are grouped to perform the Machine Learning Task as Follows.

manifesto_id	text	cmp_code	category	sentence_group
GER_202109	"We support…"	403.1	Economy	1
GER_202109	"…are taxes…"	403.2	Economy	1
GER_202109	"Democracy is essential"	201	Freedom and Democracy	2

After Grouping, the distribution of Character Lengths is checked again. The longer text length should make the Dimensionality Reduction and Clustering easier.

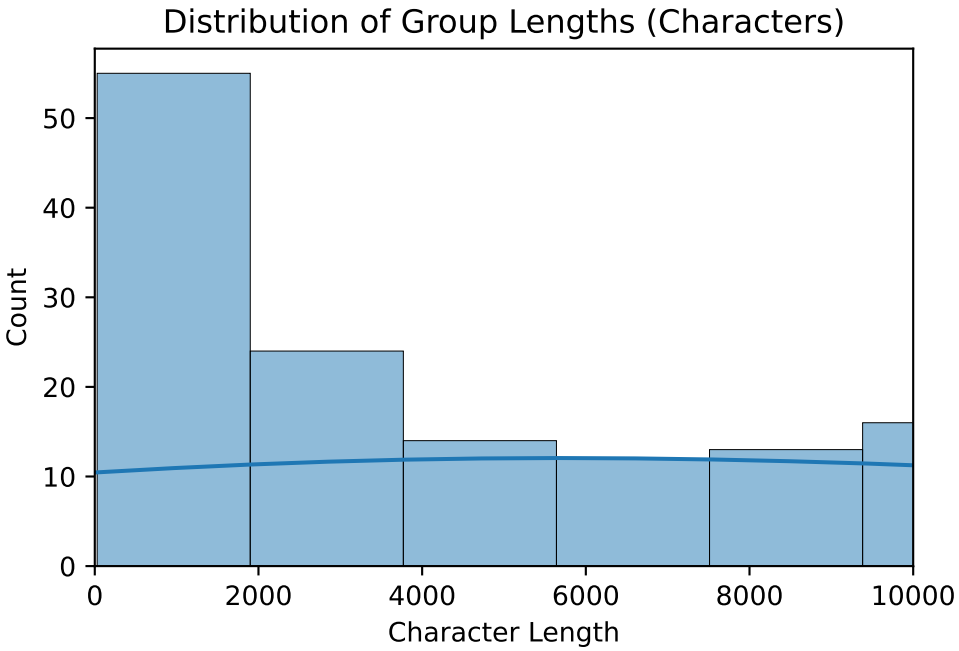
Each quasi-sentence group was then prepared for embedding using a sentence-transformer model in the next stage of the pipeline (described in Section 3).

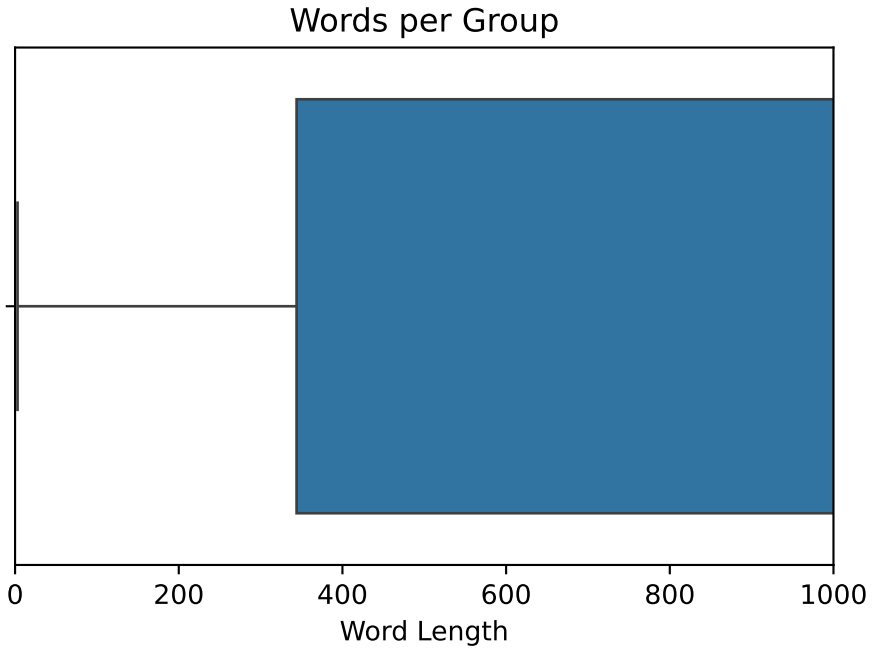
```
   manifesto_id              category  \
0  41113_202109               Economy
1  41113_202109     External Relations
```

```
2  41113_202109      Fabric of Society
3  41113_202109  Freedom and Democracy
4  41113_202109       Political System
```

```
                                                  text
0  Through science and progress. We have seen how...
1  We have learned how limited national answers t...
2  It has shown in a good way the commonality, in...
3  Dear voters, it is through elections that a so...
4  We, BÜNDNIS 90/DIE GRÜNEN, are presenting our ...
```

1.7 Data Exploration of Sentence Groups





2 Methods

The central aim of this project is to cluster quasi-sentences from political manifestos into coherent topic groups, ideally resembling the seven high-level categories defined by the Comparative Manifesto Project (CMP). Since no sentence-level topic labels are used during model training, the task is unsupervised in nature.

This section describes and motivates the methods used at each stage: transforming raw text into a numerical representation, reducing dimensionality for visualization and noise reduction, and applying clustering algorithms to identify topic-like groupings. Special attention is given to the suitability of each technique for short political text fragments and the challenges inherent to topic modeling without supervision.

2.1 Sentence Embedding

Quasi-sentences from manifestos are typically short—often no more than a clause or single policy statement. To represent these as inputs for clustering, the project uses *sentence embeddings*, which map each sentence to a fixed-length vector in a high-dimensional space.

The selected model is **Sentence-BERT (SBERT)**, a transformer-based architecture specifically designed for semantically meaningful sentence embeddings. Unlike vanilla BERT, which isn't optimized for sentence-level similarity, SBERT introduces a siamese network

structure that enables efficient semantic similarity computation. The pre-trained model `all-MiniLM-L6-v2` was chosen for its balance between performance and speed.

Sentence embeddings are preferred over traditional bag-of-words or TF-IDF vectors for several reasons:

- They capture **semantic similarity**, not just token overlap.
- They produce **fixed-size dense vectors**, suitable for distance-based clustering.
- They perform well on short, syntactically diverse texts such as political quasi-sentences.

Each quasi-sentence is thus encoded into a 384-dimensional vector, which serves as the input to subsequent analysis.

2.2 Dimensionality Reduction

Before applying clustering algorithms, the embedding space is optionally reduced to a lower dimension to address two issues:

- **Curse of dimensionality**: In high-dimensional spaces, distance metrics become less meaningful, which can degrade clustering performance.
- **Visualization**: Human interpretation of cluster structure requires 2D or 3D projections.

The method of choice is **Principal Component Analysis (PCA)**. PCA projects the data onto a lower-dimensional orthogonal subspace that captures as much variance as possible. In this project, PCA is used primarily for:

- Reducing noise in the input to clustering
- Projecting cluster centroids for manual inspection
- Identifying dominant axes of variation across quasi-sentences (e.g., economy vs. welfare focus)

PCA is chosen over nonlinear methods such as t-SNE or UMAP because:

- It is **deterministic** and interpretable.
- It preserves **global structure** better, which is useful for clustering.
- It enables visualization of loadings (important features) via **biplots**.

2.3 Clustering Algorithms

2.3.1 K-Means Clustering

The first method applied is **k-means clustering**, a centroid-based algorithm that partitions the data into k clusters by minimizing intra-cluster variance. The objective function is:

$$\underset{C}{\operatorname{argmin}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in C_i} \|\mathbf{x}_j - \mu_i\|^2$$

where μ_i is the centroid of cluster C_1.

K-means is a natural choice because:

- It scales efficiently to large datasets.
- It provides **hard assignments** (each sentence belongs to exactly one cluster).
- It requires minimal assumptions about data distribution.

However, k-means has notable limitations:

- It assumes **spherical clusters of similar size**, which may not hold for natural language data.
- It is sensitive to initialization and the chosen number of clusters k.

To mitigate this, the **elbow method** and **silhouette scores** are used to identify a suitable value for k, with $k = 7$ being a theoretically motivated target corresponding to the CMP categories.

2.3.2 Hierarchical Agglomerative Clustering

To complement k-means, **hierarchical agglomerative clustering (HAC)** is applied. HAC recursively merges the most similar clusters based on a linkage criterion until all data points are grouped into one hierarchy (a dendrogram). The project uses **Ward's method**, which minimizes the increase in total within-cluster variance when merging clusters.

HAC offers several advantages in this context:

- It does not require pre-specifying the number of clusters.
- It can reveal **nested structure**, which is conceptually useful since some CMP topics (e.g., Economy and Welfare) may be subtopics of broader ideological themes.
- It provides a visual dendrogram, which helps in interpreting inter-topic relationships.

HAC is particularly suitable for this task due to the lack of strong assumptions about cluster shape and the ability to **explore multiple granularities** of topic clusters by cutting the dendrogram at different heights.

2.3.3 Evaluation Metrics

Since this is an unsupervised task, standard accuracy metrics are not applicable. Instead, clustering quality is evaluated using:

- **Silhouette Score**: Measures how well-separated the clusters are.
- **Intra-cluster vs. Inter-cluster Distances**: Helps evaluate cohesion and separation.
- **Qualitative Inspection**: The top sentences closest to each cluster centroid are examined for thematic coherence and alignment with CMP macro categories.

If partial labeled data is available (e.g., from hand-coded samples), **Adjusted Rand Index (ARI)** or **Normalized Mutual Information (NMI)** may also be computed for external validation.

The combination of SBERT for semantic representation, PCA for structure reduction, and both k-means and hierarchical clustering for discovery provides a robust pipeline for the unsupervised categorization of manifesto text. Each method was selected to balance interpretability, computational feasibility, and alignment with the nature of political language data.

3 Implementation and Results

This section describes the practical application of the methods to the manifesto quasi-sentences. The steps include generating sentence embeddings, reducing dimensionality for structure and visualization, applying clustering algorithms, and evaluating clustering performance. Finally, the resulting clusters are analyzed to address the political research question: **Do socialist parties emphasize welfare and quality of life more often than right-wing populist parties?**

3.1 Sentence Embedding

To convert each quasi-sentence in `manifestos_df["text"]` into a numerical vector, we use the pre-trained `all-MiniLM-L6-v2` Sentence-BERT model. Each sentence is encoded into a 384-dimensional vector capturing semantic similarity.

```
Daten aus Cache geladen.
```

A few example embeddings:

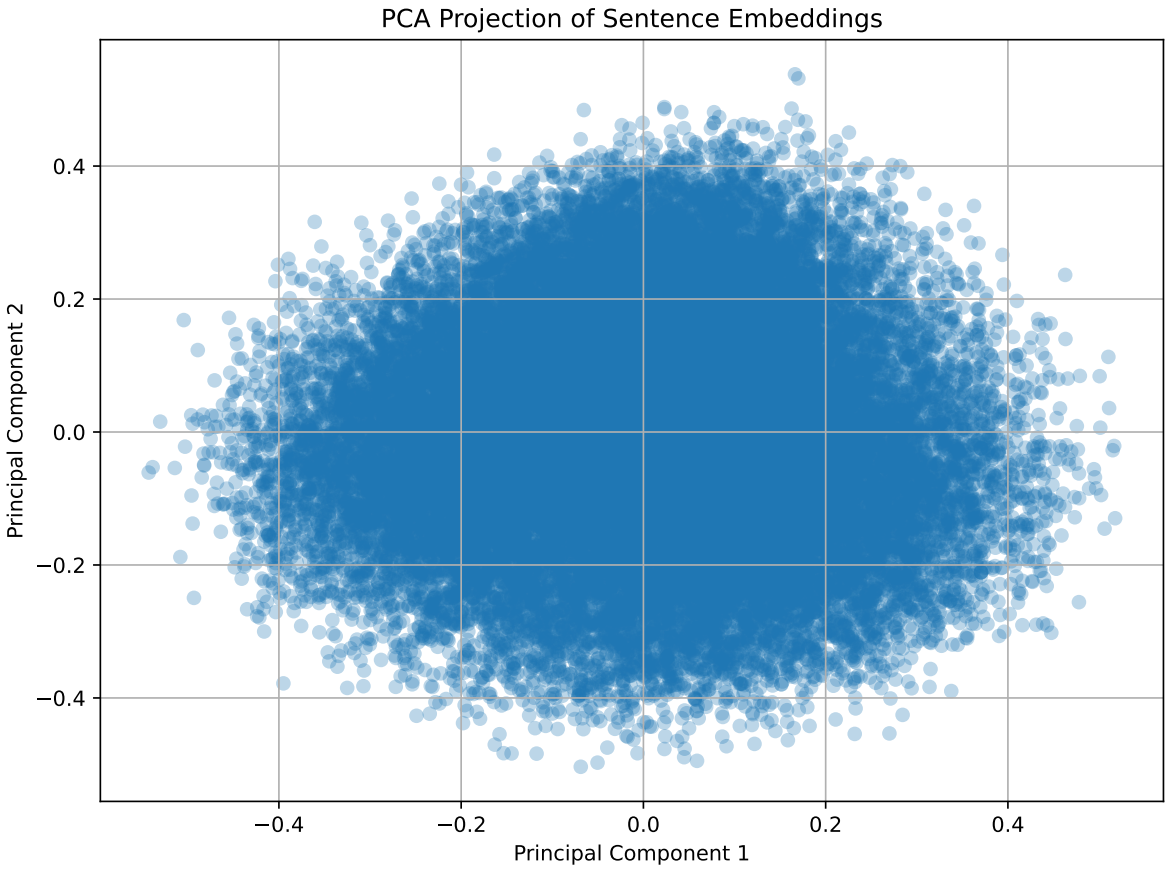
	dim_0	dim_1	dim_2	dim_3	dim_4	dim_5	dim_6	dim_7	dim_8	dim_9	...	dim_374	d
0	0.020657	-0.010217	0.013811	0.011986	0.011888	0.000365	-0.039776	-0.029985	-0.013995	0.			
1	0.030968	-0.047497	-0.054813	-0.012154	0.077466	0.004317	-0.063012	0.071940	-0.084385	-0			
2	0.104659	-0.061780	0.027408	0.065094	0.054538	-0.057555	0.000832	0.048637	0.057563	-0			

These vectors serve as the input to both the dimensionality reduction and clustering steps.

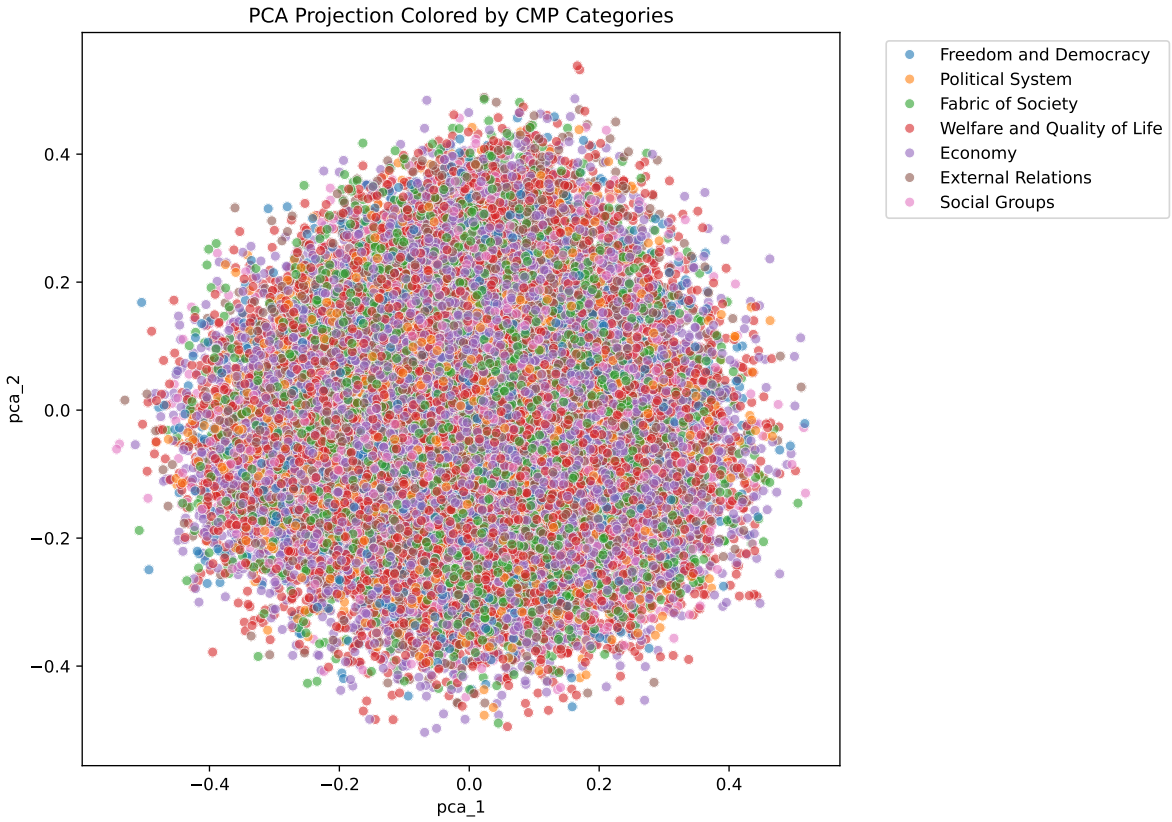
3.2 Dimensionality Reduction

To visualize the sentence embeddings and reduce potential noise, we apply **Principal Component Analysis (PCA)** to project the high-dimensional space into two dimensions.

```
PCA-Embedding aus Cache geladen.
```



To understand how the PCA dimensions relate to the underlying CMP topics, we can color the points by the manually annotated categories.



The visualization shows some topic separation, although there is noticeable overlap — expected given the complexity and brevity of the quasi-sentences.

3.3 Clustering and Evaluation

We apply two clustering algorithms: **k-means** and **hierarchical agglomerative clustering** (Ward linkage). Since we are working toward seven CMP topics, we set the number of clusters to `k = 7`.

3.3.1 K-Means Clustering

Evaluate clustering using Adjusted Rand Index (ARI), which measures similarity between the clustering and the known `category` labels (ignoring label names).

```
K-Means-Clustering aus Cache geladen.
Adjusted Rand Index (K-Means): 0.070
```

3.3.2 Hierarchical Clustering

```
Hierarchical Clustering aus Cache geladen.
Adjusted Rand Index (Hierarchical): 0.074
```

3.3.3 Cluster Composition and Topic Alignment

To better understand what each cluster contains, we look at representative sentences closest to the cluster centroids.

```
Cluster 0 Representative:
The reasons for this disadvantage may vary; but pension fund contributions, which increase wi
--------------------------------------------------------------------------------
Cluster 1 Representative:
We will increasingly integrate preventive controls into antitrust law.
--------------------------------------------------------------------------------
Cluster 2 Representative:
Minimum sentences of ten years for sexual offenses
--------------------------------------------------------------------------------
Cluster 3 Representative:
We cannot predict what leeway the state will have after Corona.
--------------------------------------------------------------------------------
Cluster 4 Representative:
We have to do everything we can to keep it that way.
--------------------------------------------------------------------------------
Cluster 5 Representative:
The passenger car toll was a disaster waiting to happen.
--------------------------------------------------------------------------------
Cluster 6 Representative:
Extending political rights to all residents
--------------------------------------------------------------------------------
```

This reveals interpretable topics per cluster, which are manually compared to the seven CMP categories for qualitative alignment.

Cluster 0 = Cluster 1 = Cluster 2 = Cluster 3 = Cluster 4 = Cluster 5 = Cluster 6 =