Data Literacy 2025 Project Report

Ansel Cheung [* 1] **Alessio Villa** [* 2] **Bartol Markovinović** [* 3] **Martín López de Ipiña** [* 4] **Niklas Abraham** [* 5]

Abstract

Cultural narratives encode and transmit evolving societal values, yet quantifying how meanings change over time remains methodologically challenging. This project investigates semantic evolution in cinema by analyzing how genres and thematic clusters shift within a unified semantic space across multiple decades. By representing movies as embeddings and tracking their geometric trajectories measuring velocity, acceleration, and curvature we distinguish periods of gradual semantic drift from moments of structural reorganization in cinematic history. This framework provides a quantitative foundation for understanding cultural change at scale and tests whether established linguistic laws of semantic evolution extend to film as a cultural medium.

1. Introduction

Cinema provides a rich archive of narrative structures that encode evolving societal values across generations. Stories serve not only to entertain but to instruct, and those narratives that align with existing social values are more likely to survive and propagate through collective memory. Recent computational work has revealed hidden cultural patterns in large narrative corpora. (Xu et al., 2020) used word embeddings to uncover systematic gender stereotypes in movie synopses, revealing the "Cinderella complex" where female characters' happiness depends asymmetrically on male characters. (Matthews & Glitre, 2021) applied topic modeling to investigate genre structure and temporal evolution, demon-

strating that lexical features capture meaningful genre conventions and showing how genres shift in composition over time. These studies establish that quantitative methods can illuminate cultural phenomena at scales beyond traditional close reading, revealing patterns that operate across thousands of narratives.

However, measuring semantic change in cultural narratives over extended historical periods remains methodologically challenging. Previous approaches have examined genre structure at specific moments or through discrete topic models that capture lexical shifts but not the continuous geometric evolution of semantic categories. Can we characterize not merely that genres change, but how they change whether through gradual drift, sudden discontinuities, or cyclical patterns? Furthermore, while linguistic corpora have been analyzed for semantic drift using diachronic word embeddings, these methods require temporal alignment procedures that introduce potential artifacts when comparing meanings across decades.

We address these questions by constructing a unified semantic space from a large corpus of film plot summaries spanning multiple decades. By embedding all narratives into a single static vector space, we eliminate temporal alignment requirements while preserving fine grained semantic relationships. Within this space, we represent genres and thematic clusters as centroids and track their trajectories over time. By computing geometric properties of these trajectories including velocity, acceleration, and curvature we can distinguish periods of continuous semantic evolution from moments of structural reorganization where genres undergo fundamental conceptual shifts. This geometric analysis reveals not just that meanings change, but the dynamics of how they change, providing quantitative measures of cultural evolution.

2. Data and Methods

2.1. Data Collection

We constructed our movie corpus using a multi stage pipeline that systematically integrated three complementary sources: Wikidata, The Movie Database (TMDb), and Wikipedia. This approach combines rich structured metadata with the detailed textual content required for semantic

---

[*]Equal contribution [1]Matrikelnummer 7274374, MSc Machine Learning [2]Matrikelnummer 7306912, MSc Computer Science [3]Matrikelnummer 7324790, MSc Machine Learning [4]Matrikelnummer 7293076, MSc Machine Learning [5]Matrikelnummer 7307188, MSc Machine Learning. Correspondence to: Initials1 <ansel-heng-yu.cheung@uni-tuebingen.de>, Initials2 <alessio.villa@student.uni-tuebingen.de>, Initials3 <bartol.markovinovic@student.uni-tuebingen.de>, Initials4 <martin.lopez-de-ipina-munoz@student.uni-tuebingen.de>, Initials5 <niklas-sebastian.abraham@student.uni-tuebingen.de>.

analysis.

Initial dataset was constructed by querying Wikidata for movies released from 1930 to 2024. In order to adhere to Wikidata's query size limitations, we iterated through the years and first acquired QIDs of all Wikidata items which have a Wikidata class that is an indirect subclass of film and have a first publication date in the given year. During this step we removed QIDs of items that do not have an English Wikipedia page associated with them. We also tried to remove non-feature movies by excluding subclasses of classes "short film" and "television series episode". However, this filtering was not perfect and further filtering of Wikidata classes was performed during post-processing. After acquiring the list of identifiers, we processed them in small batches of 20 and queried Wikidata for each movie's features including title, release date, duration, genres, directors, actors, English Wikipedia link, and very importantly links to external movie databases TMDb and IMDb. Additionally, box office, box office currency, budget and budget currency values were also queried, but they had very low coverage in the raw dataset and were not used in the final analysis.

Second, we enriched the dataset using TMDb, a community driven database that offers quantitative measures of popularity and user engagement. Wikidata's external identifiers enabled direct mapping to TMDb entries, from which we programmatically retrieved vote counts, vote averages, and popularity metrics for each film. These measures served as proxies for audience engagement and cultural impact, informing downstream film filtering and weighting.

The third stage, the most data intensive, focused on obtaining full text plot summaries. Leveraging Wikipedia sitelinks from Wikidata, we accessed each film's Wikipedia page to extract the plot section. Wikipedia's editorial standards ensure relatively uniform and neutral plot descriptions, facilitating standardized comparative semantic analysis. This step used the Wikipedia API for article retrieval, section extraction, and text normalization, transforming metadata into the dense textual data required for downstream embedding.

The last enrichment stage addressed limitations in TMDb voting statistics. Many movies in the corpus lacked TMDb ratings or vote counts, and when available, these counts were often substantially lower than those reported by other platforms. To address this issue, we enriched the dataset with IMDb vote averages and vote counts, obtained from IMDb's non-commercial data files and merged using the IMDb title identifier. The inclusion of IMDb data ensures broader coverage and higher vote volumes, resulting in a more stable measure of audience reception.

All data sources are open and appropriately licensed. Wikidata (Wikimedia Foundation, 2024a) is released under CC0 1.0 Universal (public domain). Wikipedia (Wikimedia Foundation, 2024b) is under CC BY SA 4.0, and TMDb (TMDb, 2024) under CC BY NC 4.0, allowing non commercial research with attribution. This ensures reproducibility and legal compliance.

Our final dataset achieves high coverage, with plot summaries for over 80% of films in most decades (see Table 1).

2.2. Data cleaning

After collecting the raw movie data from Wikidata, TMDb and Wikipedia, we first ensured that our dataset does not contain any duplicates with respect to Wikidata QIDs and Wikipedia links. Then we performed the following data filtration and cleaning steps:

- **Filtering out movies without a Wikipedia plot.**

- **Removal of non-feature movies.** We removed samples from our dataset that had a Wikidata class that is an indirect subclass of a class that does not describe a feature movie. Some examples of not feasible Wikidata classes include trailers, television series episodes, short films and radio programs.

- **Filtering out movies with excessively long plots.** We filtered out movies with plots longer than 14,000 characters from our dataset because these plots are labeled by Wikipedia as *excessively long*.

- **Removal of movies with low entropy plots.**

- **Genre filtering.** We filtered out genres that appear only once in the dataset because these genres obviously do not describe a group of movies.

- **Exclusion of explicit content.** We excluded movies with pornographic and exploitation genres. WHY???

The most critical cleaning step was the removal of movies with low-entropy plots. Raw dataset contained a significant number of incomplete or overly brief plots (e.g. this). To identify and remove such movies we employed a filtering method inspired by (?), who used perplexity of a Large Language model to filter out low quality documents. While (?) utilized perplexity of a 5-gram language model trained on high quality data, we tokenized the plots with the BGE-M3 tokenizer and compute the Shannon entropy of the token distribution for each plot. To determine the optimal entropy threshold, we sampled 150 movies from the borderline entropy region of $[4.0, 5.5]$ and manually annotated them as either *good* or *bad* quality. The threshold of 4.8398 was chosen to maximize the $F\beta$ score with $\beta = 0.5$ prioritizing precision over recall.
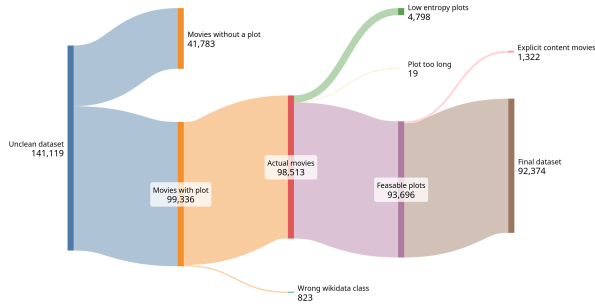
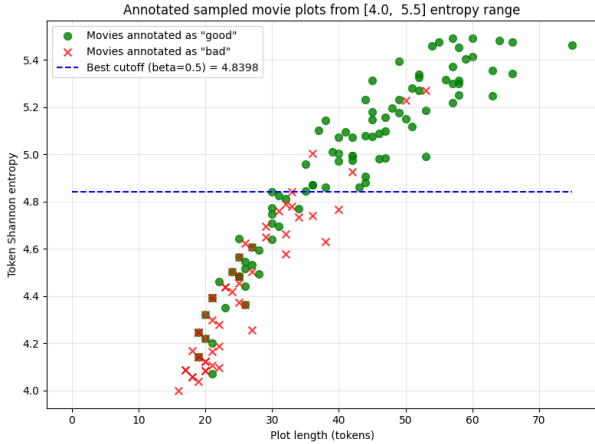*Figure 1.* Data cleaning pipeline: number of movies retained after each filtering step.

Figure 2. Results of manual labeling of 150 plots in the borderline entropy region with the chosen entropy threshold

2.3. Novelty analysis

This part in introduction maybe: Common public sentiment is that the film industry is "running out of ideas" resulting in movies that are becoming less creative and more similar to each other over time.

Then later: To investigate the claim that movies are becoming less novel over time, we developed a metric for novelty defined as the minimal cosine distance between a specific movie's plot embedding and the embeddings of all movies in the dataset released prior to it. This can be formally written as:

$$\text{Novelty}(m_i) = \min_{j:year_j < year_i} \left(1 - \frac{E(m_i) \cdot E(m_j)}{\|E(m_i)\| \|E(m_j)\|} \right) \tag{1}$$

where $E(m)$ denotes the embedding vector of a movie's plot. Intuitively, a higher novelty score indicates that the movie's plot is more dissimilar from prior movies, while a lower score implies existence of a very similar movie released earlier. To compiute these scores, we use the Faiss library

(**?**). Movies were sorted chronologically and processed in yearly batches. For a given batch we queried the Faiss index containing all prior movies to find distances to the nearest neighbor for each movie in the current batch. After that, the current batch was added to the index for subsequent queries.

Results: In order to assess if temporal trends of novelty scores exist, we plot the average novelty score per year alongside scattered individual movie scores in Figure 3.
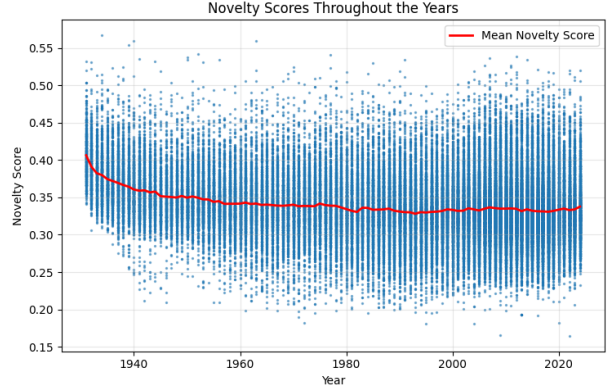

Figure 3. Novelty scores of movies over time. The blue line represents the average novelty score per year, while individual movie scores are shown as scattered points.

The resulting plot indicates that the average yearly novelty remained relatively constant from 1950s onwards. TODO: Find better way to plot this, maybe novelty of all movies vs novelty of Oscar nominees?

Table 1. Per decade feature coverage (%) of key metadata fields in the movie dataset.

Decade	Actors+Director	Genre	Plot	Vote Count
1950s	86.56	63.67	83.03	90.19
1960s	83.82	61.01	77.31	85.26
1970s	86.55	62.47	79.57	86.58
1980s	84.68	59.41	79.80	85.41
1990s	82.22	58.33	82.00	84.49
2000s	77.34	60.51	84.19	83.73
2010s	70.25	60.55	84.94	85.60
2020s	70.66	64.85	77.59	89.63
Average	80.26	61.35	81.05	86.36

After the data was collected in a tabular format, the textual plot descriptions required transformation into vector representations via a suitable embedding model for downstream analysis. The plot descriptions extracted from Wikipedia pages exhibit substantial variability in length, ranging from 10 to 20,479 characters, corresponding to approximately 6 to 5,296 tokens in a english tokenizer. All plot descriptions in our corpus are in English, which simplifies the embedding

process by eliminating cross lingual considerations.

2.4. Methodology

After the data was collected and cleaned, the first step was to embed the movie plot summaries into a semantic space.

The selection of an appropriate embedding model was guided by the Massive Text Embedding Benchmark (MTEB) leaderboard results[1], which provides comprehensive evaluations of embedding models across diverse retrieval and semantic similarity tasks. Based on these benchmarks, we selected the BGE-M3 (Beijing Academy of Artificial Intelligence Multilingual, Multifunctional, Multi-granularity) model (Chen et al., 2024), an open-source model developed by the Beijing Academy of Artificial Intelligence. The BGE-M3 model achieved competitive performance (28th place on the MTEB leaderboard) while maintaining a relatively compact architecture with 0.5 billion parameters. Critically, the model supports a context length of 8,192 tokens, which enables embedding entire movie plot descriptions into a single vector representation without requiring chunking.

The BGE-M3 model provides three embedding modes: a dense vector (global document representation via [CLS]), a sparse vector (high-dimensional token weights for lexical matching), and a multi-vector mode (token-level semantic representations for each word in the input).

A key methodological choice was to use a single, static embedding model for all time periods, rather than training separate or temporally aligned models. This ensures all film plots are represented in a unified latent space, avoiding complex post-hoc alignment and minimizing artifacts. While movies from earlier decades describe very different world view with a different language and culture, the Wikipedia-based plot summaries are not contemporaneous texts from those eras, instead, they are modern English descriptions collectively maintained and updated since Wikipedia's founding in 2001. Thus, any linguistic variation or semantic drift in the summaries themselves is minimal. We rely on this assumption of consistent descriptive language to enable direct, meaningful comparisons of embedding-based semantics across decades, without additional alignment steps.

Embedding variable-length documents presents challenges for transformer-based models due to fixed context windows and representational biases. Because plot summaries in our dataset span from a few sentences to thousands of words, it is crucial to select chunking and pooling strategies that minimize length bias while retaining semantic content. Relying solely on the [CLS] token for global representation can introduce substantial bias: it is sensitive to input length, often overemphasizes the first 128 tokens (Devlin et al.,

2018; Gong et al., 2019), and underperforms mean pooling on long documents (Raffel et al., 2023). With over 75% of our plots exceeding 512 tokens, a more robust aggregation method is required to avoid discarding information.

There were four main document embedding methods evaluated: (1) Mean Pooling: compute the average of token embeddings to form the document vector; (2) Early Chunk-then-Embed: split the document into chunks before embedding and average their vectors, reducing length effects but limiting cross-chunk context; (3) Late Embed-then-Chunk: embed the full document, then aggregate over sliding windows to capture broader context; (4) CLS Token: use the pretrained [CLS] token as a summary. Each method offers a different tradeoff between bias, variance, and preservation of semantic details.

Once movie plots are embedded into a unified semantic space, quantitative analysis of their geometric relationships becomes possible through distance metrics. The cosine distance between embeddings provides a natural measure of semantic dissimilarity, enabling the construction of cumulative distribution functions over pairwise distances within defined subsets of the corpus. Such distributions encode structural properties of the embedding space and reveal whether semantic relationships exhibit systematic patterns across temporal periods or thematic categories.
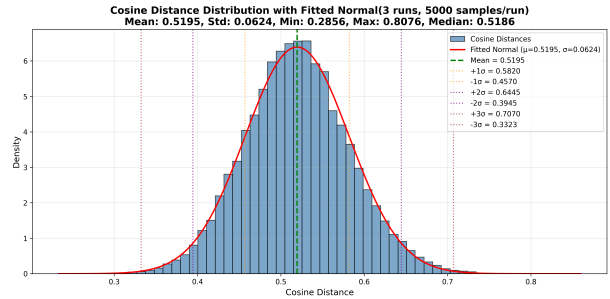

Figure 4. Cosine distance distribution with fitted normal distribution. The distribution shows the pairwise cosine distances between movie embeddings, with mean $\mu = 0.5195$ and standard deviation $\sigma = 0.0624$. The histogram represents cosine distances from 3 runs with 5000 samples each, demonstrating the approximately normal structure of semantic relationships in the embedding space.

To rigorously compare distance distributions across different subsets of movies, for instance films from different decades or belonging to distinct genres, we employ the Kolmogorov-Smirnov test, a non parametric statistical method for assessing whether two empirical distributions arise from the same underlying continuous distribution (Massey, 1951). The two sample KS test compares the empirical cumulative distribution functions (ECDFs) of two samples by computing the maximum vertical distance between them.

Formally, given two samples $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$,

their empirical cumulative distribution functions are defined as:

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n}1_{X_i \leq x}, \quad G_m(y) = \frac{1}{m}\sum_{j=1}^{m}1_{Y_j \leq y} \quad (2)$$

where 1 denotes the indicator function. The KS test statistic is defined as the supremum of absolute differences between these ECDFs:

$$D_{n,m} = \sup_x |F_n(x)G_m(x)| \quad (3)$$

Under the null hypothesis that both samples are drawn from the same continuous distribution, the distribution of $D_{n,m}$ is known and can be used to compute p values for hypothesis testing. The test is particularly suited for our application because it makes no assumptions about the underlying distributional form, is sensitive to differences in both location and shape, and operates directly on the distance measurements without requiring binning or parametric modeling.

A key design choice in applying this framework is the selection of a reference point from which distances are computed. One natural approach is to use the mean vector of a baseline subset of movies as a reference embedding, then compute the distribution of distances from this reference point to all movies in the corpus. This enables quantitative analysis of how semantic representations are spatially organized relative to fixed reference points in the embedding space.

In the context of temporal semantic analysis, the KS test enables systematic comparison of distance distributions across decades. By computing distances from fixed reference points (such as mean embeddings of genre clusters) to movies from different decades, we can assess whether the spatial organization of semantic representations evolves over time. If the semantic structure of cinema remains stable over time, distance distributions should remain statistically similar. Conversely, significant differences in these distributions, as detected by the KS test, would indicate structural reorganization of the semantic space, suggesting periods where narrative conventions undergo fundamental shifts. This analysis quantifies how the density and dispersion of semantic representations evolve temporally, providing a complementary perspective to trajectory based analyses of genre evolution.

2.5. Genre analysis

As movie genres provide a meaningful taxonomy with potential temporal evolution patterns, we examine semantic drift across different time periods. To this end, embeddings are first grouped by genre g into discrete time periods τ, forming the set $\mathcal{M}_g^{(\tau)}$ of plot embeddings. For each group, two alternative representative embeddings are computed:

the **centroid** (arithmetic mean) $\bar{\mathbf{e}}_g^{(\tau)}$ and the **medoid** (cosine distance minimizer embedding) $\tilde{\mathbf{e}}_g^{(\tau)}$.

With an arbitrary number of years Δt per group, the period index τ is calculated by flooring the movie year to the nearest multiple of Δt:

$$\tau = \left\lfloor \frac{\text{year}}{\Delta t} \right\rfloor \cdot \Delta t \quad (4)$$

We computed the following metrics to analyse the drift dynamics across the groups:

Genre drift and acceleration: drift (Equation 5) measures displacement between representative embeddings of consecutive periods, capturing how a genre's semantic center evolves over time. Acceleration quantifies the change in drift between consecutive periods.

$$\mathbf{d}_g^{(\tau)} = \bar{\mathbf{e}}_g^{(\tau+\Delta t)} - \bar{\mathbf{e}}_g^{(\tau)} \quad (5)$$

Inter genre distance: determines cosine distance between representatives of each pair of genres for each year, enabling pairwise comparison between specific genres.

Due to group size differences between time periods, two alternative normalization approaches have been employed: (1) downsampling, ensuring equal sampling error across groups, and (2) z-score normalization, which accounts for the standard error of the difference between group means:

$$\hat{v}_g^{(\tau)} = \frac{v_g^{(\tau)}}{\sigma_{\text{pooled}} \cdot \sqrt{\frac{1}{n_g^{(\tau)}} + \frac{1}{n_g^{(\tau+\Delta t)}}}} \quad (6)$$

where σ_{pooled} is the pooled within group standard deviation of cosine distances, and $n_g^{(\tau)}$ is the number of movies in genre g at time τ.

None of the genre based analyses yielded statistically significant results, suggesting that genres may be too broad as analytical categories and any underlying patterns are likely obscured by noise. Figure 5 illustrates temporal drift for the three most popular genres, computed over 5-year periods. Each period was downsampled to 145 movies, with 95% confidence intervals estimated via bootstrapping (1,000 samples).

Projection along temporal axis This idea is to select an arbitrary vector that is in the embedding space and project the movie embeddings onto this vector. Based on the choice of this vector, we will be able to see how much of each movie embedding lies on the vector, and we can then do temporal analysis to see how this metric evolves over time. Since our embeddings are already normalized to magnitude 1, cosine distance is proportional to L2 norm, which measures euclidean distance between embeddings. Note that for this
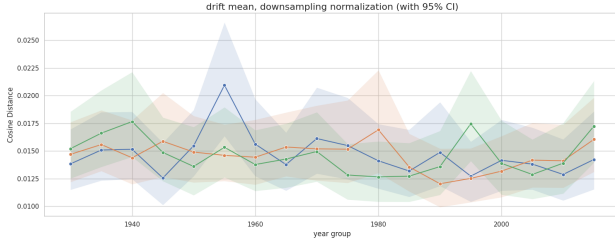
Figure 5. Genre drift from 1930 to 2025 of three most popular genres over 5-year periods

Projection Analysis, all experiments were bootstrapped for 500 times, 1000 samples.
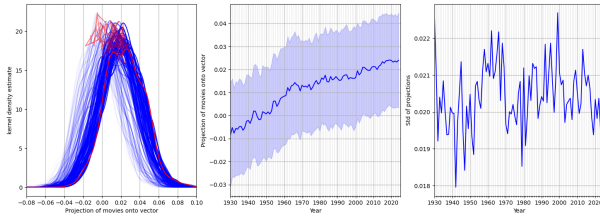

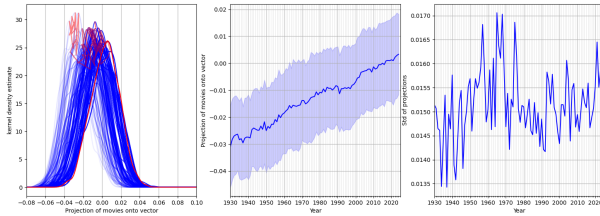Figure 6. Projection onto action vector over years

Figure 7. Projection onto Romance vector over years

First vector we chose was $mean(\text{emb}_{\text{action 2024}}) - mean(\text{emb}_{\text{action 1930}})$. From Figure 6, we can see that movie plots are increasingly becoming similar to action movie plots. If we look at Figure 7 we also see the same trend for Romance. This created suspicion and we plotted the same thing but the overall centroid shift from 1930 to 2024 and also saw the same plots (Figure 8). This could mean that how movie plots evolve over time outweighs the shift in genres.

In order to then explore the shift in genres over time, we plotted the Cosine Distance evolution of mean action embedding per year from the overall centroid. From Figure 9 we observe that the action movies are getting more and more similar to the average embedding vector. The Romance cosine distance to the same mean vector shows a different story (Figure 10).

Figure 11 We expanded this to our new genres and this revealed that many large genres were converging towards the
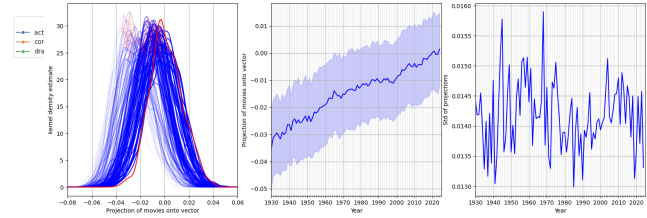

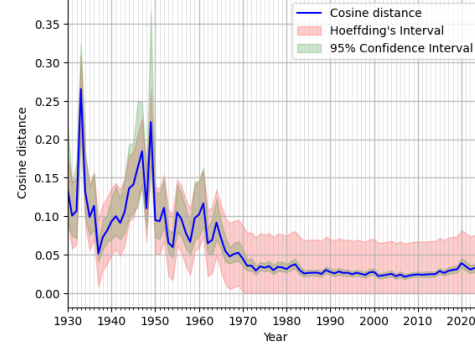Figure 8. Projection onto Mean vector over years

Figure 9. Action cosine distance to mean (all) embedding vector

mean embedding. There are some outliers such as "Anime" which only appeared after 1980. "Film Noir" and "Adventure and Fantasy" did not follow the trend of converging towards the mean embedding. (do I end here?)

Spread analysis Another way to see if movies are converging over the years or spreading out is to measure the spread per year. (https://arxiv.org/pdf/1810.08693) Frobenius norm measures the total variance of each year's difference in movie embeddings to its yearly mean embedding. The frobenius norm (https://arxiv.org/pdf/1501.01571 page 84) is the sum of squared singular values in which we are only measuring noise and not the signal. In order to see the signal shift, we use the spectral norm to find the maximum singular value of the difference in movie embeddings and their respective yearly mean embeddings.

We observe from Figure 12 that movies are getting getting more and dissimilar. Paired together with the mean L2 distance from each movie to the yearly mean embedding, we can see that the average distance from each movie to its yearly mean remains the same, but the spectral norm triples in size (Figure 13), signalling that there is some sort of shape shift or stretching of the embedding cloud. There is a sharp drop in 2020, which might be attributed to movie production during covid period reducing the number of movies being produced and hence reducing the chance for outliers.

The next sensible step to take is to analyze the explained variance of the first principal component. This can simply
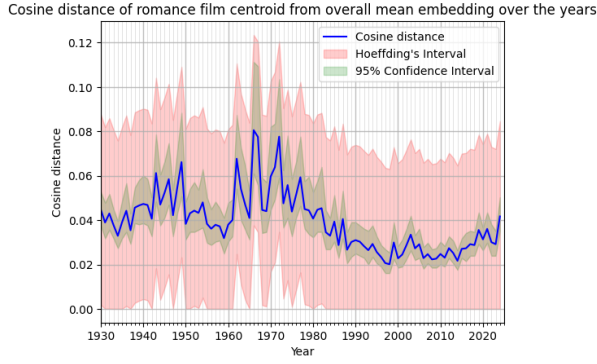
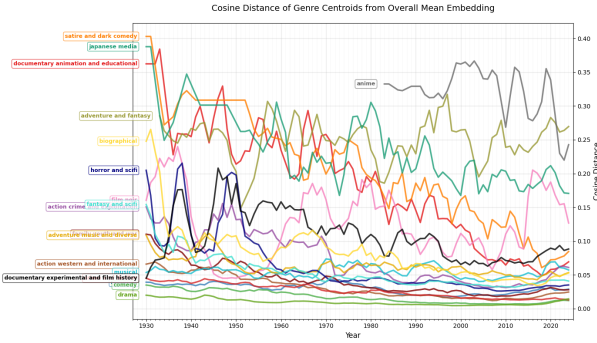*Figure 10.* Romance cosine distance to mean (all) embedding vector

Figure 11. All genre's cosine distance to mean (all) embedding vector

Figure 12. Mean L2 Norm vs Cosine Distance against yearly centroid

Figure 13. Spectral and Frobenius Norm of yearly movies against yearly centroid

be calculated by the squared spectral norm divided by the squared frobenius norm. From Figure 14, the explained variance increases from 3% to 4% and sharply rises to 4.75% after 2020. An explained variance of 4% is significant in a dataset with dimension of 1024. If all dimensions were random noise, each PC would explain $1/1024 \approx 0.098\%$, so 4% is 40 times higher than random. This means that there is a direction which is polarizing the movie industry.

This is actually one large circular path leading back to PCA. Now we can interpret the evolution of movies to see which movies are the most polarized by that particular year's principal component. Hence we project every movie's embeddings onto its year's PC. From here we select an arbitrary number of years to analyze its top and bottom 5 movies.

(PCA1 table here)
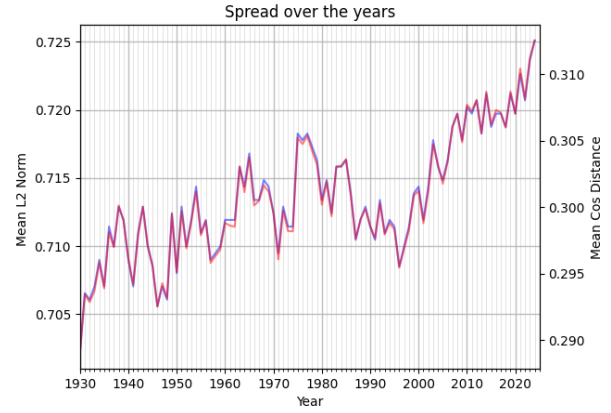
(PCA US Movies)

(PCA German Movies)

3. Results

3.1. Chunking Method Comparison

We compared ten embedding strategies on 5,000 movie plots, based on four core approaches: Mean Pooling, CLS Token, Chunk-First Embed (three chunk/stride combinations), and Late Chunking (six configurations). Evaluation metrics included length bias, isotropy, genre classification, and class separation. Table 2 summarizes the main results.

The evaluation reveals that no single method dominates across all metrics. Length-norm correlation varies substantially: MeanPooling and LateChunking show strong positive correlation (0.62 to 0.82), while ChunkFirstEmbed exhibits negative correlation (down to -0.37). CLS Token achieves near-zero correlation (0.004), effectively removing length bias.

Isotropy also differs across approaches. MeanPooling and LateChunking have higher first-PC variance (11.3-11.9%),

Table 2. Chunking method performance. Length Norm Corr: correlation of document length and embedding norm. Isotropy (1st PC): variance in first principal component. Silhouette: cluster cohesion. Sep. Ratio: intra/inter class similarity.

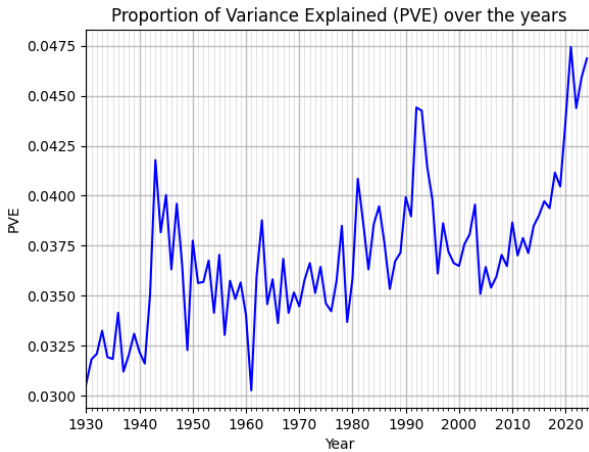| Method	Length-Norm Corr	Isotropy (1st PC %)	Silhouette	Sep. Ratio
Mean Pooling	0.629	11.36	-0.036	0.943
CLS Token	0.004	3.32	-0.016	0.958
Chunk-First 512/256	-0.366	3.47	-0.020	0.951
Chunk-First 1024/512	-0.275	3.37	-0.017	0.959
Chunk-First 2048/1024	-0.031	3.33	-0.016	0.961
Late Chunk 512/256	0.822	11.92	-0.037	0.949
Late Chunk 1024/512	0.726	11.53	-0.037	0.948
Late Chunk 2048/1024	0.656	11.36	-0.036	0.958
Late Chunk 2048/512	0.656	11.36	-0.036	0.958
Late Chunk 512/0	0.821	11.40	-0.037	0.948

Figure 14. PC1 Explained Variance of yearly movies against yearly centroid

suggesting less uniform embeddings, while CLS Token and ChunkFirstEmbed achieve lower values (3.3-3.5%), indicating better isotropy. For genre classification, accuracy varies minimally (0.326 to 0.349), with ChunkFirstEmbed_1024_512 performing best, though improvements are marginal. Silhouette scores remain negative across all methods, indicating substantial genre overlap in embedding space.

Separation ratios range from 0.943 to 0.961, with CLS Token achieving 0.958, among the highest values. Cosine similarity distributions show consistent means (0.485-0.505), but CLS Token and ChunkFirstEmbed produce more concentrated distributions (std: 0.061-0.067) compared to pooling-based methods (std: 0.123-0.126).

Given that metrics vary without a clear winner across all dimensions, we select CLS Token as our embedding strategy. CLS Token is the most common and simple approach, requires no parameter tuning, and offers strong separability (separation ratio of 0.958) while maintaining minimal length bias and good isotropy properties.

4. Literature and Methods Review

4.1. Semantic Representation and Cultural Patterns

Our use of word embeddings to represent narrative content draws significant inspiration from the work of *Xu et al. (2019)*. In their study of the "Cinderella Complex" the authors demonstrated that high-dimensional vector spaces can capture latent social biases and emotional dependencies within movie synopses. While *Xu et al.* focused on the internal dynamics of characters and gender stereotypes, our research shifts the lens toward the **inter-textual relationships** between films. By embedding Wikipedia plots, we treat the narrative as a holistic semantic unit, allowing us to map the entire "cinematic universe" into a geometric space where thematic similarity is measured by vector proximity.

4.2. Quantifying Novelty and Innovation

To define and measure "novelty" we adopt the theoretical framework proposed by *Sreenivasan (2013)*, who defines cultural innovation as the emergence of atypical combinations of elements. However, our methodology diverges from Sreenivasan's reliance on keyword frequencies and probabilistic modeling. Instead, we utilize **Geometric Distance Analysis**. In our model, novelty is operationalized as the distance of a movie vector from its k-nearest neighbors in the embedding space. This allows for a more nuanced detection of innovation: a film is considered "novel" if its semantic coordinates lie in a sparse region of the vector space, indicating a narrative structure that is mathematically distant from established conventions.

4.3. Diachronic Evolution and Genre Trajectories

The temporal dimension of our analysis is informed by the diachronic architectures discussed by *Hamilton et al. (2016)*. Their research into how word meanings shift over time provides a blueprint for our analysis of genre evolution. By calculating the **centroid** of specific genres across different decades, we track their "semantic drift." This approach allows us to test hypotheses regarding cultural homogenization versus diversification, a concept explored in the musical domain by *Di Marco et al. (2025)*. While *Di Marco et al.* utilized network science to observe the simplification of musical structures, we apply these concepts to text, observing whether movie genres are converging toward a standard "formula" or expanding into new, unexplored territories of the embedding world.

5. Discussion & Conclusion

Contribution Statement

References

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., and Liu, Z. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024. URL https://arxiv.org/abs/2402.03216.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Gong, L., He, D., Li, Z., Qin, T., Wang, L., and Liu, T. Efficient training of BERT by progressively stacking. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2337–2346. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/gong19a.html.

Günther, M., Mohr, I., Williams, D. J., Wang, B., and Xiao, H. Late chunking: Contextual chunk embeddings using long-context embedding models, 2025. URL https://arxiv.org/abs/2409.04701.

Massey, F. J. The kolmogorov–smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46 (253):68–78, 1951.

Matthews, P. and Glitre, K. Genre analysis of movies using a topic model of plot summaries. *Journal of the Association for Information Science and Technology*, 72(12):1511–1527, 2021. doi: 10.1002/asi.24525.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL https://arxiv.org/abs/1910.10683.

TMDb. The movie database (tmdb). https://www.themoviedb.org, 2024. Licensed under CC BY-NC 4.0 for non-commercial use.

Wikimedia Foundation. Wikidata. https://www.wikidata.org, 2024a. Licensed under CC0 1.0 Universal (Public Domain).

Wikimedia Foundation. Wikipedia, the free encyclopedia. https://www.wikipedia.org, 2024b. Licensed under CC BY-SA 4.0.

Xu, H., Zhang, Z., Wu, L., and Wang, C.-J. The cinderella complex: Word embeddings reveal gender stereotypes in

movies and books. *PLOS ONE*, 15(e0225385), 2020. doi: 10.1371/journal.pone.0225385.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489, San Diego, California, jun 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1174. URL https://aclanthology.org/N16-1174/.