Plot Holes and Text Topology

Stanford CS224N Custom Project (Grading Option 3)

Egor Alimpiev
Department of Mathematics
Stanford University
alimpiev@stanford.edu

Vivek Myers
Department of Mathematics
Department of Computer Science
Stanford University
vmyers@stanford.edu

1 Introduction and Related Work

A *word embedding* is a map that takes a word from some (usually fixed) vocabulary to a point (vector) in an embedding space, usually taken to be a high-dimensional Euclidean space $\mathbb{R}^d$. Word embeddings are essential in most of NLP problems and usually serve as a first stage in neural models for named-entity recognition [1], sentiment analysis [2], language modeling [3], text summarization [4] and question answering [5], word-level machine translation [6], coreference resolution [7]. The main assumption underlying the algorithms for word embedding construction is that the word's sense is captured by surrounding words within the text. So, even though all word embeddings use the context at *training time*, some of the models, such as word2vec [8] and GloVe [9] output the same vector for a word regardless of its context (we call these embeddings *non-contextual*), and others require the context (usually a sentence in which a word occurs) to produce a vector in the embedding space. These *contextual* methods are ELMo [10] and BERT [11].

Apart from being extremely useful in practice, word embeddings can be a subject of study in their own right. Understanding the nature of the embedding space would be a step towards more interpretable and traceable neural models for NLP. One of the first phenomena discovered was "word vector arithmetic" [12] that showed that word2vec can capture semantic relations of English language. Other studies continued using *local* approaches, such as examining linear algebraic structure of word vectors [13] to show that embeddings for polysemous words correspond to linear combinations for "virtual" vectors for each word sense. In [14], the authors concentrated on studying the neighborhood of a word vector with a goal of solving word sense disambiguation. Of course, many papers, such as [15], concentrated on practical aspects such as determining the best loss function and dimensionality to optimize training and reduce overfitting. In is important to notice that all of these papers work with *non-contextual* embeddings.

Any attempt to manually inspect the embedding space is obstructed by the high number of dimensions (usually $50 \leq d \leq 768$) and the fact that any dimensionality reduction method, such as MDS, PCA, SVD, t-SNE, U-MAP, etc., necessarily distorts the picture and loses some information.

In this paper, we propose to take a global approach to studying word embeddings, one that would apply to any particular embedding algorithm and be invariant with respect to the dimensionality of the embedding space. We attempt to extract *topological* summaries and interpret them in the light of known results.

We are following the paradigm of Topological Data Analysis (TDA), a relatively young field that tries to apply methods of algebraic topology to real world data. TDA was introduced by Gunnar Carlsson, a professor of mathematics at Stanford University [16, 17]. The reason for the development of TDA was the nature of real world data: in most cases, the data we have is in the form of a point cloud in a high-dimensional space, the data is noisy, the metrics and coordinates used are not natural (in a sense that thay rarely carry intrinsic meaning). Using toplogical methods allows us to, first of all, extract both quantitative and qualitative insights into the dataset and then incorporate topological information into statistical and machine learning pipelines. TDA was successfully applied in virology [18] and medicine [19] to infer evolutionary patterns and identify subsets of data of particular interest, in image analysis [20] and imaging in physics [21], materials science [22], etc. The challenge in

many of these works is to provide a good interpretation of the topological invariants in the context of a particular application. We discuss questions of interpretation below.

2 Approach

Given some text (corpus), word embedding models produce a point cloud in an embedding space $\mathbb{R}^d$, with one point for each token of the text. We then interpret this point cloud as a (possibly noisy) sample from some underlying subspace of $\mathbb{R}^d$ and thus reduce the study of the text embedding to studying the topology of the (hypothetical) underlying space.

Our main computational approach is *persistent homology*. We define persistent homology rigorously in the next subsection, but, intuitively, it counts the number of "holes" in different dimensions (*connected components* in dimension 0, *loops or tunnels* in dimension 1, *voids* in dimension 2 and so on) at varying level of resolution. In mathematics, more "holes" in more dimensions is naturally associated with increased complexity, as simplest most understood spaces such as balls do not have any "holes" and are *contractible*.

Computing persistent homology over a range of text allows us to (mostly qualitatively) measure of the extent of nonhomogeneity of the embedding space. Nonuniformity of the embedding space must reflect the internal structure and "power" of the embedding mechanism, the amount of information contained in a word vector. This is because a random choice of word vectors – the least powerful method – would produce uniform embeddings with no internal structure. This connection between topological complexity as outputted by our computations and intrinsic power of word embeddings is the basis for all of the interpretations that follow.

In particular, we concentrated on the following questions:

(a) It is well known that models based on contextual embeddings outperform those using non-contextual ones. Do we see this on the level of topology, thathis, does the topological complexity differ between the two groups of methods?

(b) Does the topological complexity differ across different categories of texts: we consider different writing styles (poetry, prose, news articles, and scientific papers) and different readability levels (as measured by Dale-Chall readability scale).

(c) What are the dynamics of large-scale topology of the text embedding as it is processed by a contextual embedding model?

As usual, we begin with definitions.

2.1 Persistent Homology

Persistent homology is based on *simplicial homology*, and a concise definition is in Appendix A. There we also provide a small pictorial example of simplicial homology calculation.

The crucial step in TDA is to convert the data point cloud into a simplicial complex. This is achieved by a construction called a *Vietoris-Rips complex*.

Definition 1 *For a set of points $V = \{v_1, \dots, v_n\} \in \mathbb{R}^n$ and a fixed positive number $r > 0$, a Vietoris-Rips (VR) complex at scale r $\mathbf{VR}_r(V)$ is defined as having V as a set of vertices (0-simplices) and a k-simplex for each set of k vertices of v of diameter less than r. That is, for each set of k vertices such that each pairwise distance is less than r.*

In applications, the dimensions of the complex is limited. We consider simplices of dimension at most 3.

Notice that when r is increasing, new simplices can appear but existing ones can not disappear: $\mathbf{VR}_{r_1}(V) \subseteq \mathbf{VR}_{r_2}(V)$ if $r_1 \leq r_2$. This way, for some set of points $V = \{v_i\} \in \mathbb{R}^d$ and a sequence of scales $0 < r_1 < \cdots < r_l$ we can form a *filtered* simplicial Vietoris-Rips complex $\mathbf{VR}(V) = \{\mathbf{VR}_{r_i}(V) \mid i = 1, \dots, l\}$.

Figure 1a shows an example of a filtered VR complex. It is important to notice that some of the holes are "noise", artifacts that depend on the sample, and some correspond to the actual shape of the annulus

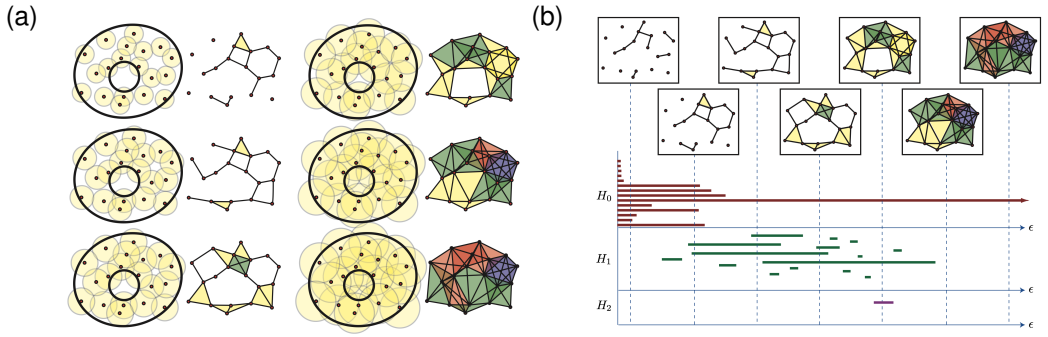


Figure 1: (a) Vietoris-Rips simplicial complex at various scales for a sample of points representing an annulus. [17, Figure 3] (b) An example of barcode plots for the VR complex for a sample from an annulus. We can obtain Betti numbers by considering "vertical slices" at a particular scale, counting the number of intervals that intersect them. [17, Figure 4].

The homology of the filtered complex is now a sequence of homology groups $H_k(\mathbf{VR}_{r_i}, R)$ at each scale r_i. The collection of homology groups of a filtered complex in every dimension is called *persistent homology*.

2.2 Persistence Diagrams

We can extract numerical data from persistent homology of data. It is usually called a *persistence barcode*.

Definition 2 *Persistence barcode for dimension k is a multiset of intervals that mark births and deaths of k-dimensional holes detected by kth homology group as the scale r of the filtration increases.*

One can view these intervals as tracking how Betti numbers (Definition 8) change as the scale grows.

The set of barcodes for every dimension can be either plotted as a set on horizontal intervals with the x-axis being the filtration scale – a *barcode plot*, or as a set of points on a rectangular plot where the x-axis is the time of birth of the hole (beginning point of the interval) and the y-axis is the time of death (endpoint of the interval) – a *persistence diagram*. The structure of these plots becomes more clear with some examples on Figure 1b. In what follows we present persistence diagram which incorporated data for all dimensions with points of different colors. Appendix B shows another example of persistent homology computation and compares barcode plots with persistence diagrams.

2.3 Distances

Persistence barcodes can be quantified and compared. First of all, there are naive measures such as the number of intervals in each dimensions and their mean lifetime. But, there are also distance metrics on the space of barcodes. Theoretically, for two persistence barcodes X and Y, a metric d on $\mathbb{R}^2$ and $p \in [1, \infty]$, one considers *Wasserstein distance* [23] – an infimum over all partial bijections ϕ from X to Y:

$$W_p[d](X, Y) = \inf_{\phi: X \to Y} \left[\sum_{x \in X} d(x, \phi(x))^p \right]^{1/p}, \; p \in [1, \infty)] \tag{1}$$

$$W_p[d](X, Y) = \inf_{\phi: X \to Y} \sup_{x \in X} d(x, \phi(x)), \; p\infty] \tag{2}$$

In practice, we consider *computable* values of parameters: *bottleneck distance* $W_\infty[L_\infty]$ and *sliced Wasserstein distance* [24] which is an approximation of $W_2[d]$ for d a standard Euclidean metric on $\mathbb{R}^2$.

2.4 Computations

Our computations follow the pipeline in [23]. We use the Ripser Python package [25] and produce the plots using `persim` Python package.

2.5 Baseline

As a baseline, we consider *clustering* methods and *silhouette scores* [26] that measure the extent to which the data fits its clusters. Zeroth homology group "counts" the number of connected components akin to clustering, however, we claim that using persistent homology calculations gives us significantly more information than just clustering: it allows us to work with *connected data* and make finer distinctions of shape than just number of "groups" or "lumps" of data. However, we can try to make a fair comparison by computing silhouette scores for a varying number of clusters, similar to how we vary the scale of the filtration of a VR complex.

2.6 Texts and Embeddings

General descriptions of word2vec, GloVe, ELMo, and BERT algorithms can be found in corresponding references. We use pretrained models available online: 300-dimensional word2vec vectors trained on the Google News dataset[1], 256-dimensional ELMo model available from AllenNLP [27], Base BERT (768-dimensional), uncased, available from Huggingface Transformers package [28].

GloVe embeddings [9] were used in two versions with different vocabulary sizes: with $400K$ word vocabulary trained on Wikipedia and Gigaword 5, and with $2.2M$ vocabulary size trained on the Common Crawl dataset, both 300-dimensional. This is done to assess the effect of vocabulary size on the topology of the embedding space.

We limit the text length at around 1000 word for computational reasons.

3 Experiments

3.1 Data

We conduct silhouette and persistent homology analysis on several small text corpora of ~1000 words described in Appendix C. These corpora correspond to science, news, and literature, in addition to several texts with varying Dale-Chall readability scores [29].

We also compare the results of silhouette and persistent homology analysis on Shakespeare's *Antony and Cleopatra* [30]. We conduct these analysis on both the full text as well as the text with the last scene of the last act removed.

We perform these computations for the embeddings described in Section 2.6. We perform persistent homology computations up to dimesion 3. For computational reasons, we replace our embedding for the above step with a representative subsample of 300 points selected using greedy permutation [31]. We also perform silhouette calculations for up to 100 clusters generated by k-means.

Additionally, we perform silhouette and persistent homology analysis on 6,950 corpora, consisting of articles in the 20 News Group dataset [32] and poems on the Poetry Foundation [33].

On this larger set of corpora, for computational reasons, we only compute 20 silhouette scores corresponding to the 20 values of $k \in \{2, 7, 12, \ldots, 97\}$.

3.2 A Motivating Example: Plot Holes

Given that persistent homology roughly measures the number of n dimensional "holes" in an embedding, it is enlightening to examine the effect of artificially creating a hole in the embedding of a corpus. We compare the persistent homology of the embeddings of Shakespeare's *Antony and Cleopatra* with and without the last scene to see if the "plot hole" created by its removal is topologically significant.

[1] Google News word2vec link.

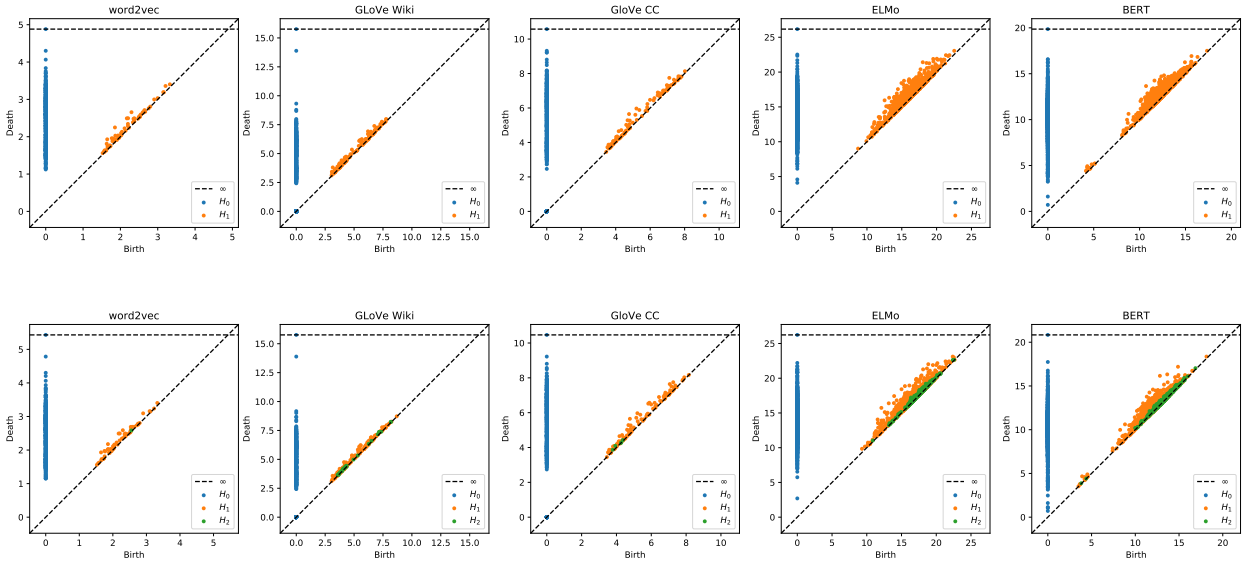


Figure 2: Comparison of persistent homology of embeddings of Shakespeare's Antony and Cleopatra: full text (top) and text with last scene removed (bottom).

Discussion Indeed, adding the "plot hole" causes numerous 2 dimensional holes to appear in the persistent homology plots. In particular, the contextual embeddings show long-lived 2-cycles when the "plot hole" is added that were previously absent.

3.3 Homology Lifetimes

We compare the mean liftime of n-cycles in the persistent homology of various word embeddings for the text corpora in Appendix C.

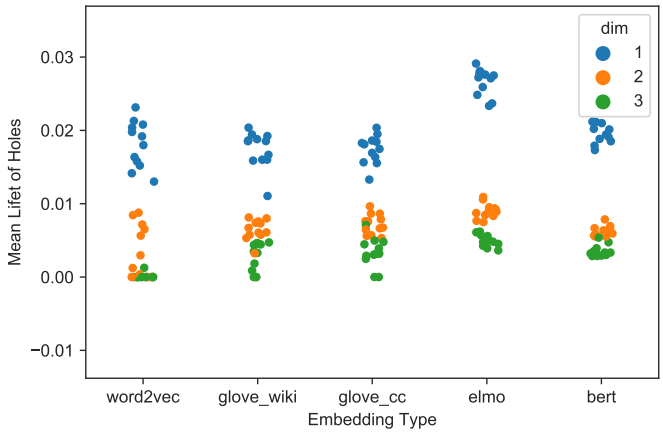


Figure 3: Comparison of average lifetime of n-cycles by embedding type, varying the dimension n.

Discussion Dimensionality of cycles is inversely related to their lifespan. The lifespans are similar for all embeddings for dimensions 2 and 3. For dimension 1, the contextual embeddings (ELMo, BERT) have the longest cycle lifespan, suggesting the contextual embeddings have the most complex 1-dimesional structure.

3.4 Analysis of Dale-Chall Readability

In this section, we analyze three particular corpora: `readability_1`, `readability_2`, and `readability_3`, corresponding to sample school essays with Dale-Chall readability scores of ~5.9, ~7.3, and ~8.7 respectively. The plots are presented in Appendix D.

Discussion Under visual inspection, the silhouette plots for the `readability` corpora are similar, with the exception of the **ELMo** embedding for `readability_2`. Across embeddings, the overall contour of the silhouette plots is similar, increasing after a steep decline and tapering off near 100.

Meanwhile, the persistent homology plots show much more complicated structure for the contextual embeddings (**ELMo**, **BERT**) relative to the other methods, with many more nontrivial 3-cycles and 1-cycles/2-cycles that persist much longer. Further, `readability_2` and `readability_3`, which have much higher Dale-Chall readability scores than `readability_1`, show more complex structure in their noncontextual embeddings (**word2vec** and **GloVe**), with many more 2-cycles appearing and 1-cycles that persist for a longer period of time than in `readability_1`.

3.5 Homology and Text Category

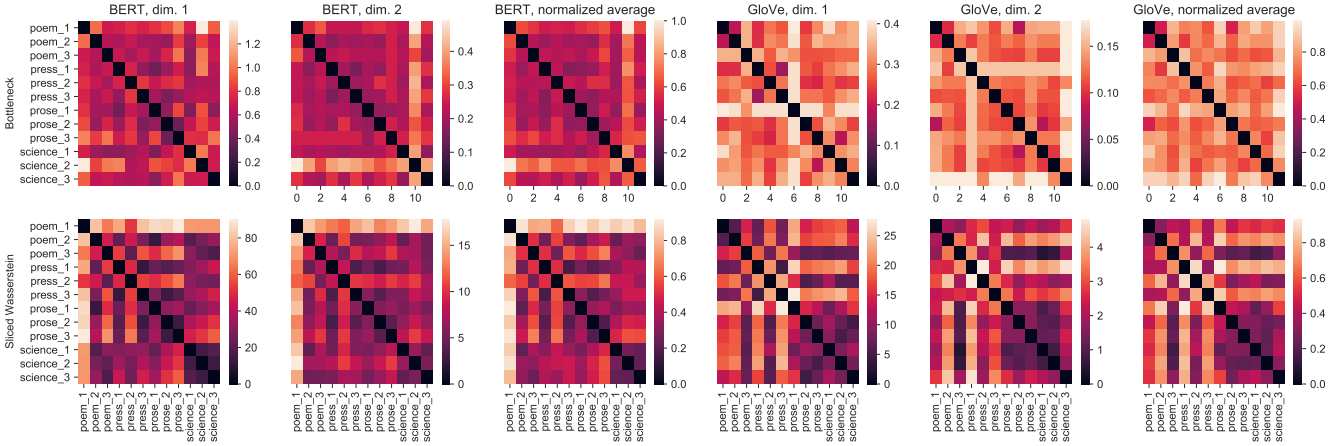


Figure 4: Bottleneck distance between persistent homology of texts in Appendix C with BERT and GloVe embeddings.

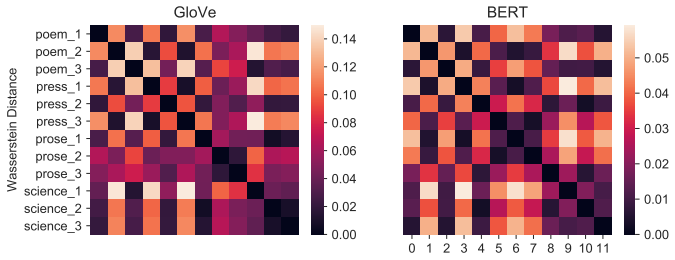


Figure 5: Wasserstein metric between silhouette scores of texts in Appendix C with BERT and GloVe embeddings.

We compare the pairwise bottleneck distances between persistent homology computations (dimensions 1 and 2) on the texts in Appendix C, as well as the pairwise Wasserstein distance between their silhouette score plots, all on the BERT and GloVe embedding schemes. The plots do not show clear evidence that diagrams of the texts of the same category are "clustered together". However, when we

compute the ratio of the average *within-topic* and *between-topic* distances (again, averaged over H_1 and H_2 for both types of distance metrics), for the Sliced Wasserstein metric it equal to ≈ 0.90 for both GloVe and BERT embeddings. Similarly, for the Wasserstein distance between silhouette score distribution, the average *within-topic* and *between-topic* distance ratio is equal to 0.86 for GloVe embeddings and 0.92 for BERT.

This suggests that persistent homology and silhouette scores can discriminate between different categories of texts, and to assess this more thoroughly, we train a classifier, as discussed in the next section.

3.6 Classifying Texts by Homology

We measure the relative importance of different persistent homology groups by looking at the accuracy of a text classifier, trained to distinguish between the articles and poems discribed in 3.1 based on a single nth persistent homology group, parametrized as a time-series of the Betti numbers. As a baseline, we also train the classifier on silhouette scores.

We encode nth persistent homology as a tensor counting the number of n cycles present at each time step discretized to 100 time steps, and enocde silhouette scores for the 20 values of k.

Our classifier is implemented using a deep 1D convolutional architecture (Figure 6). We use a learning rate of $5 \cdot 10^{-4}$ with an Adam optimizer [34] and ℓ_2 regularization with $\lambda = 10^{-3}$, with a cross entropy loss function.

For persistent homology of the Elmo embeddings for dimensions $0, 1$, and 2 as well as silhouette scores, we train the classifier to distinguish news articles and poems (described in Section 3.1). We set aside 20% of the data for validation and 20% for testing, training on 60%. We train until the exponentially weighted average of the past 20 epochs' validation loss is less than that of the past 10 epochs, meaning the validation loss is increasing.

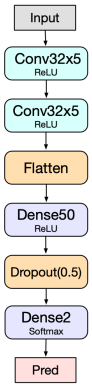


Figure 6: Architecture

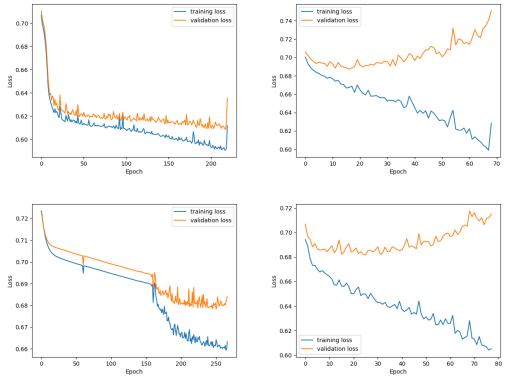


Figure 7: Training and validation loss: silhouette data (top left), H_0 (top right), H_1 (bottom left), H_2 (bottom right).

Discussion We compute the accuracy of the final trained models on the test dataset (Figure **??**). All models for persistent homology achieve greater than 50% accuracy indicating a relationship between topological structure and genre of text (news or poem). Persistent homology in dimension 2 is the strongest predictor of genre. However, silhouette scores had the strongest relationship, indicating that simple cluster analysis is at least as strong a predictor of genre as persistent homology.

Analysis	Test Accuracy
H_0	57.27%
H_1	56.91%
H_2	59.57%
silhouette	68.06%

3.7 Topology of Internal Embedding States

We present persistent homology computations on the `press_2` corpus for the BERT and ELMo embeddings (Figure 8, Figure 9).

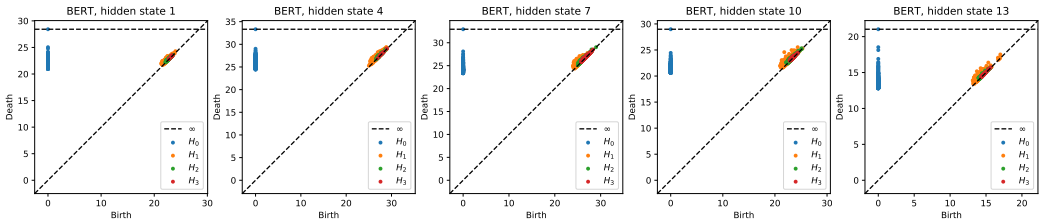


Figure 8: Bert hidden state persistent homology.

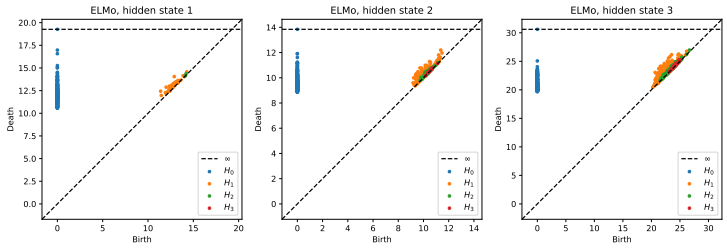


Figure 9: Elmo hidden state persistent homology.

Discussion The later hidden states show more complex topological structure for both embedding schemes. In particular, they begin to exhibit 3 dimensional structure. This can be interpreted as topological evidence for the well-known fact that as information is processed by a neural model, it moves from extracting low-level features to more high-level, more general ones.

4 Conclusion

Persistent homology computations intuitively find n dimensional holes in embeddings. We find that these computations reveal nontrivial topological structure in the word embeddings of corpora drawn from short texts, particularly for contextual word embeddings. The intuitive definition of persistent homology is consistent with its behavior on word embeddings—persistent homology can detect the addition of an artificial “plot hole" to a text by removing a key section.

For low dimensions, we demonstrate that deep models can differentiate between texts of different genres solely based on persistent homology. Thus, persistent homology computations capture structural information about texts. However, simpler methods of cluster analysis, such as silhouette scores, predict genre with a deep model at least as well as persistent homology. This result suggests that persistent homology does not capture strictly more structural information about texts (that is relevant to determining their genre) than simpler cluster analysis methods.

Instead, we can frame the advantages of using persistent homology as useful in *qualitative* assesment of embeddings. Above, using this method, we have demonstrated evidence for many "famous" results, such as: presence of high-level structure in embedding spaces that captures information about word meaning and semantics; contextual embeddings being condsiderably different and more poswerful models; increase in high-level structure of embeddings as is is processed by layers of the neural model, etc. This suggests that in the future the use of TDA methods in studying neural models may bring novel results or just serve an auxilliary method to confirm hypotheses obtained through usual and more common methods.

References

[1] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[2] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[3] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[4] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. Abstractive text summarization using sequence-to-sequence RNNs and beyond. *arXiv preprint arXiv:1602.06023*, 2016.

[5] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

[6] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

[7] Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. *arXiv preprint arXiv:1707.07045*, 2017.

[8] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[9] 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)*, pages 1532–1543, 2014.

[10] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[12] Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 746–751, 2013.

[13] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495, 2018.

[14] Tadas Temcinas. Local homology of word embeddings. *arXiv preprint arXiv:1810.10136*, 2018.

[15] Zi Yin and Yuanyuan Shen. On the dimensionality of word embedding. In *Advances in Neural Information Processing Systems*, pages 887–898, 2018.

[16] Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.

[17] Robert Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

[18] Joseph Minhow Chan, Gunnar Carlsson, and Raul Rabadan. Topology of viral evolution. *Proceedings of the National Academy of Sciences*, 110(46):18566–18571, 2013.

[19] Monica Nicolau, Arnold J Levine, and Gunnar Carlsson. Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Sciences*, 108(17):7265–7270, 2011.

[20] Gunnar Carlsson, Tigran Ishkhanov, Vin De Silva, and Afra Zomorodian. On the local behavior of spaces of natural images. *International journal of computer vision*, 76(1):1–12, 2008.

[21] Ludovic Duponchel. Exploring hyperspectral imaging data sets with topological data analysis. *Analytica chimica acta*, 1000:123–131, 2018.

[22] Takenobu Nakamura, Yasuaki Hiraoka, Akihiko Hirata, Emerson G Escolar, and Yasumasa Nishiura. Persistent homology and many-body atomic structure for medium-range order in the glass. *Nanotechnology*, 26(30):304001, 2015.

[23] Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, 2017.

[24] Mathieu Carriere, Marco Cuturi, and Steve Oudot. Sliced wasserstein kernel for persistence diagrams. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 664–673. JMLR. org, 2017.

[25] Christopher Tralie, Nathaniel Saul, and Rann Bar-On. Ripser. py: A lean persistent homology library for python. *Journal of Open Source Software*, 3(29):925, 2018.

[26] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[27] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. 2017.

[28] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.

[29] Jeanne Sternlicht Chall and Edgar Dale. *Readability revisited: The new Dale-Chall readability formula*. Brookline Books, 1995.

[30] William Shakespeare. *Anthony and Cleopatra*, volume 2. Cassell & Company, 1908.

[31] Nicholas J Cavanna, Mahmoodreza Jahanseir, and Donald R Sheehy. A geometric perspective on sparse filtrations. *arXiv preprint arXiv:1506.03797*, 2015.

[32] Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.

[33] http://www.poetryfoundation.org/. Poetry foundation.

[34] Zijun Zhang. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, pages 1–2. IEEE, 2018.

[35] Allen Hatcher. *Algebraic topology*. 2005.

A Simplicial Homology

First, we define *simplicial homology* following [35].

Definition 3 *A k-simplex is a k-dimensional polytope that is a convex hull of its $(k+1)$ vertices. A standard k-simplex is a set $S = \{(t_1, \ldots, t_{k+1}) \in \mathbb{R}^{k+1} \mid \sum_i t_i = 1, t_i > 0\}$.*

Definition 4 *A* face *of a k-simplex is a $(k-1)$-simplex that is a complex hull of any $k-1$ of the original vertices.*

Definition 5 *A simplicial complex is a set of simplices K such that every face of a simplex in K is also in K, and if two simplices $\sigma_a, \sigma_b \in K$ have a non-empty intersection, $\sigma_a \cap \sigma_b$ is a face of both σ_a and sigma b.*

Intuitively, one should imagine a collection of simplies glued together. One can define a simplicial complex by listing the vertices $\{v_0, \ldots v_m\}$ of K and then defining k-simplices as ordered k-tuples, e.g. $(v_0, \ldots, v_k)$. The ith face is then denoted as $(v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_k)$, and this is often abbreviated as $(v_0, \ldots, \hat{v}_i, \ldots, v_k)$.

Definition 6 *For a simplicial complex K, a k-chain c is a formal finite linear combination*

$$c = \sum_{i=1}^{N} a_i \sigma_i^k$$

where $\sigma_i \in K$ are k-simplices and a_i are called coefficients *and are elements of a some ring, usually $\mathbb{Z}$ or a field like $\mathbb{F}_2$.*

Notice that k-chains naturally form an abelian group under addition.

Definition 7 *A boundary map is a map d_k from a k-chain to a $(k-1)$-chain defined on simplices as*

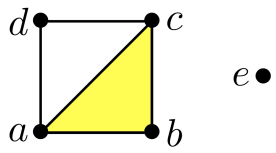
$$d((v_1, \ldots, v_k)) = \sum_{i=0}^{k} (-1)^i (v_0, \ldots, \hat{v}_i, \ldots, v_k).$$

and extended by linearity to chains. An elementary computation shows that $d_k \circ d_{k+1} = 0$ for any $(k+1)$-chain.

Now we are ready to define simplicial homology groups of a simplicial complex S. Fix a ring of coefficients R for the chains. Let $Z_k(R) = \ker d_k$, that is, a subgroup of *all* of the k-chains of S that are taken to zero by d_k, and let $B_k(R) = \operatorname{im} d_{k+1}$, that is, the image of *all* $(k+1)$-chains of S under the map d_{k+1}. Then, kth simplicial homology group of S with coefficients in R is defined as a quotient $H_k(S; R) = Z_k(R)/B_k(R)$. Notice that if the coefficients of chains were integers, $R = \mathbb{Z}$, $H_k(S; \mathbb{Z})$ is an abelian group, and if the coefficient ring is a field, for example $R = \mathbb{F}_2$, $H_k(S; \mathbb{F}_2)$ is a vector space.

Definition 8 *In a special case of field coefficients, kth Betti number $\beta_k(S)$ is defined as the dimension of $H_k(S)$.*

(a)



(b) We compute the simplicial homology for the simplicial complex in Figure 3(a). We have the following sequence of vector spaces and linear maps:

$$0 \longrightarrow \mathbb{F}_2 \xrightarrow{d_2} \mathbb{F}_2^5 \xrightarrow{d_1} \mathbb{F}_2^5 \xrightarrow{d_0} 0\,.$$

Let abc denote the basis vector that corresponds to the simplex $\{a, b, c\}$. Similarly, we use ab, ac, ad, bc, and cd to denote the basis vectors that correspond to the 1-simplices; and we use a, b, c, d, and e to denote the basis vectors that correspond to the 0-simplices. We order the bases of the vector spaces using lexicographic order. We then have

$$d_2 = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \end{pmatrix}^{\mathrm{t}}$$

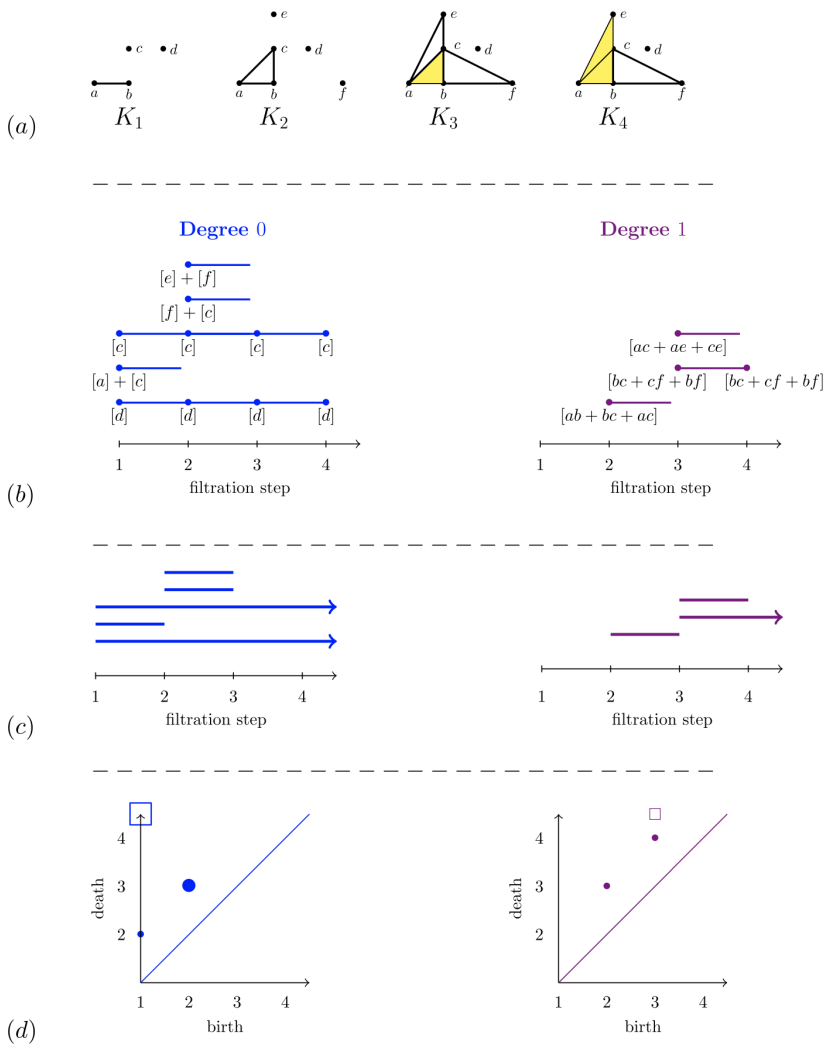
and

$$d_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

One can then compute that $\beta_0(K) = 2$, $\beta_1(K) = 1$, and all higher Betti numbers are 0.

Figure 10: (a) A simple simplicial complex, [23, Figure 3]. (b) Computation of simplicial homology for the complex in (a) with coefficients in a field $\mathbb{F}_2$, [23, Figure 4].

B Persistence Diagram (Additional Example)



This example from [23, Figure 5] illustrates the computation for a small VR complex. (a) shows the filtration steps for the complex. (b) shows the barcode plots for dimensios 1 and 2 with each hole labeled by corresponding simplices that generate it. (c) shows the same barcode plot in a conventional way with arrows indicating intervals that persist until the last filtration step. (d) shows the same information as a persistence diagram. Note that the two plots in (d) can be combined.

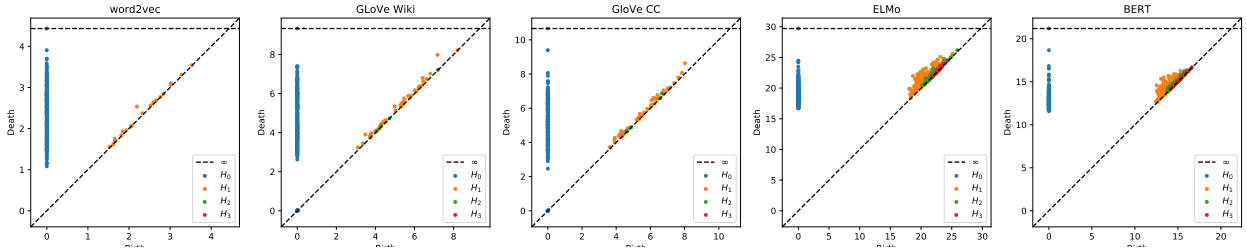
C Descriptions of Corpora

- `poem_1` — *The Lady of Shalott* by Alfred, Lord Tennyson, 980 words, obtained from poetryfoundation
- `poem_2` — *The Waste Land* by T. S. Eliot, first two parts, 1016 words, obtained from poetryfoundation
- `poem_3` — *The Highwayman* by Alfred Noyles, 1011 words, obtained from poetryfoundation
- `press_1` — *How we came to live in "cursed" times* from Newyorker, part, 1010 words
- `press_2` — *F.B.I. Opened Inquiry Into Whether Trump Was Secretly Working on Behalf of Russia* from New York Times, part, 1022 words
- `press_3` — *Three years of misery inside Google, the happiest company in tech* from The Wired, part, 1033 words
- `science_1` — *The Conley index, gauge theory, and triangulations* by Ciprian Manolescu, obtained from his homepage, part, 975 words
- `science_2` — *Attention Is All You Need* by Vaswani et al., obtained from arxiv, part, 1004 words
- `science_3` — *Deep learning: new computational modelling techniques for genomics* by Eraslan et al., from Nature Reviews Genetics, part, 1010 words
- `prose_1` — *Pride and Prejudice* by Jane Austen, obtained from Project Gutenberg, part, 1005 words
- `prose_2` — *The Great Gatsby* by F. Scott Fitzgerald, obtained from Progect Gutenberg Australia, part, 1001 words
- `prose_3` — *The Hobbit* by J. R. R. Tolkien, obtained from Internet Archive, part, 1024 words
- `readability_1` — a sample school essay obtained from Thoughtful Learning, 487 words, Dale-Chall readability score ~5.9
- `readability_2` — a sample school essay obtained from Thoughtful Learning, 1018 words, Dale-Chall readability score ~7.3
- `readability_3` — an entry on *Moral Relativism* from Stanford Plato, 1007 words, Dale-Chall readability score 8.7

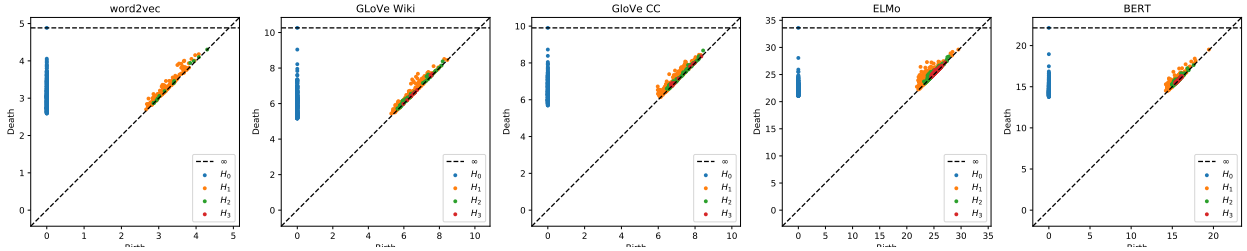
D Readability Scale Plots

D.1 Persistent Homology

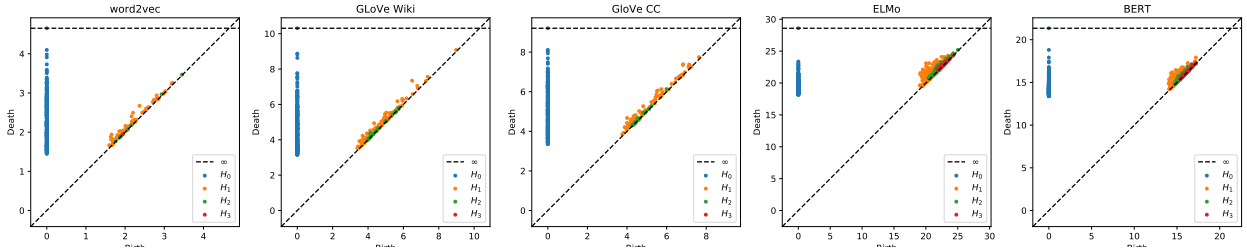
- `readability_1:`



- `readability_2:`

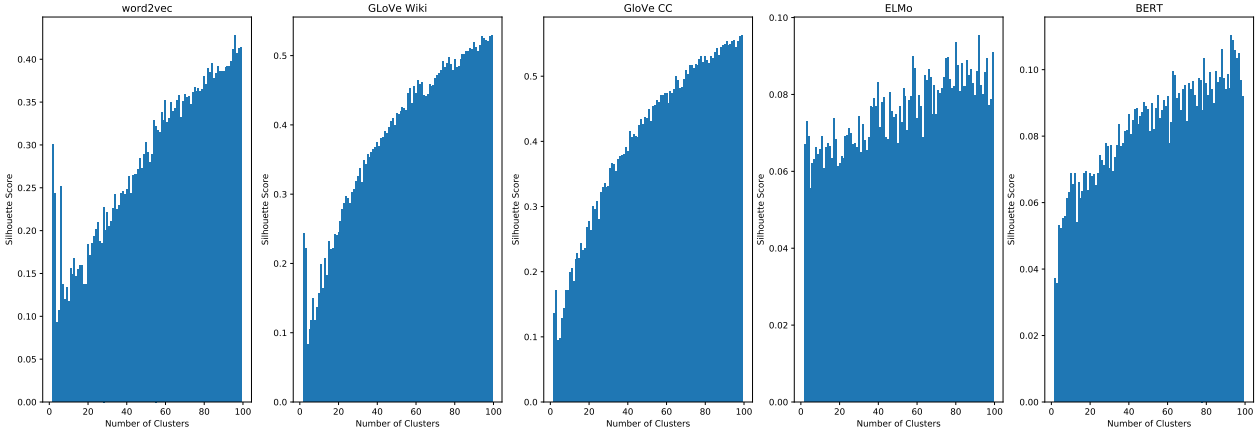


- `readability_3:`

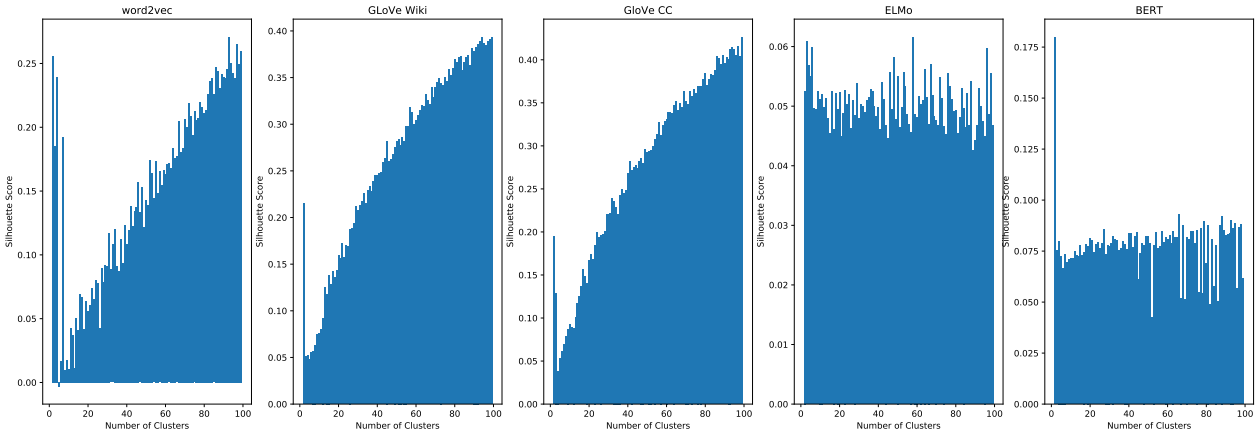


D.2 Silhouette Score

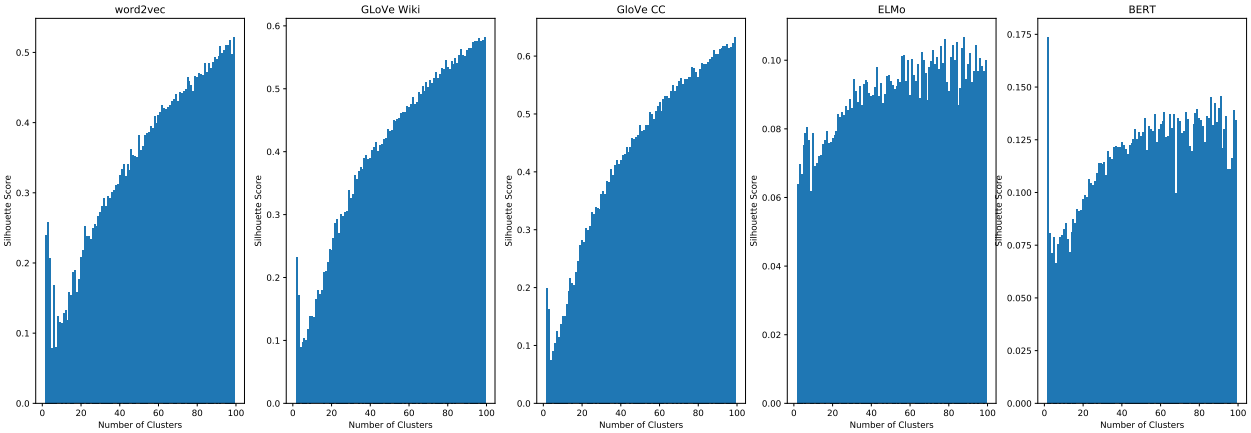
- `readability_1:`



- `readability_2:`

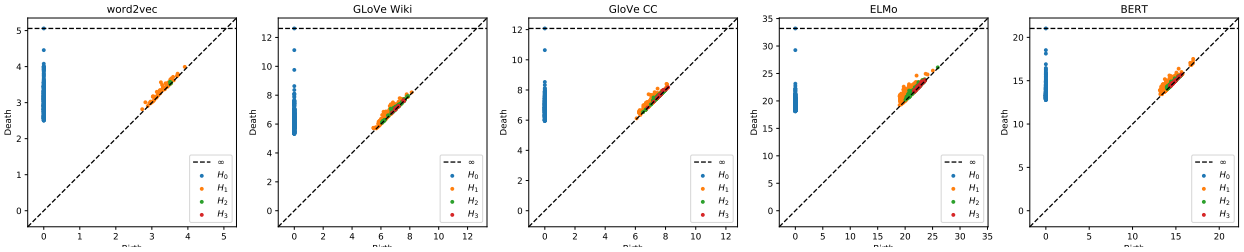


- `readability_3:`

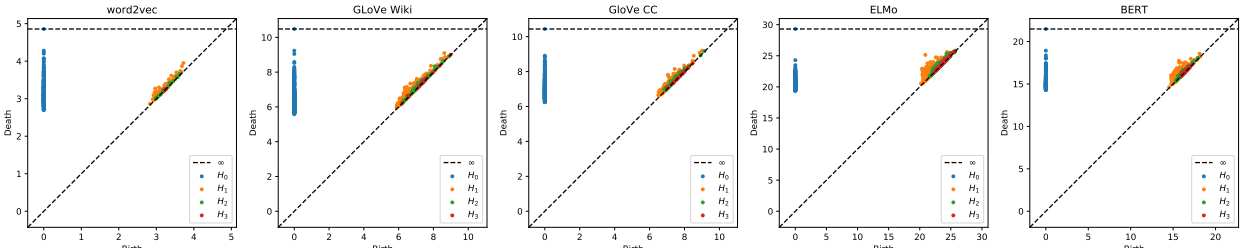


E Other Persistent Homology Plots

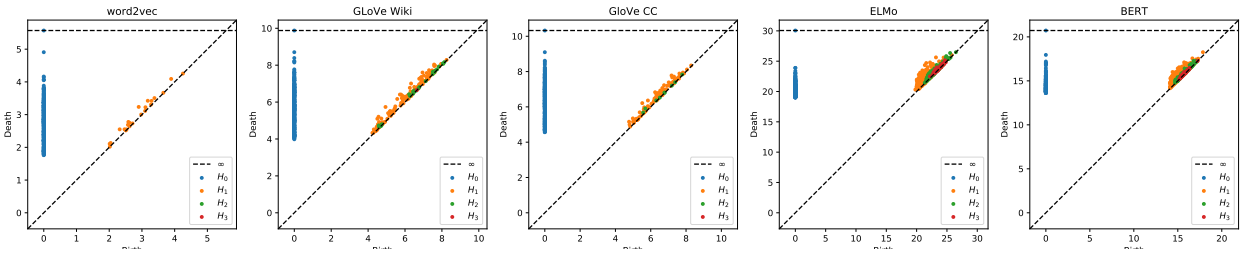
- `poem_1:`



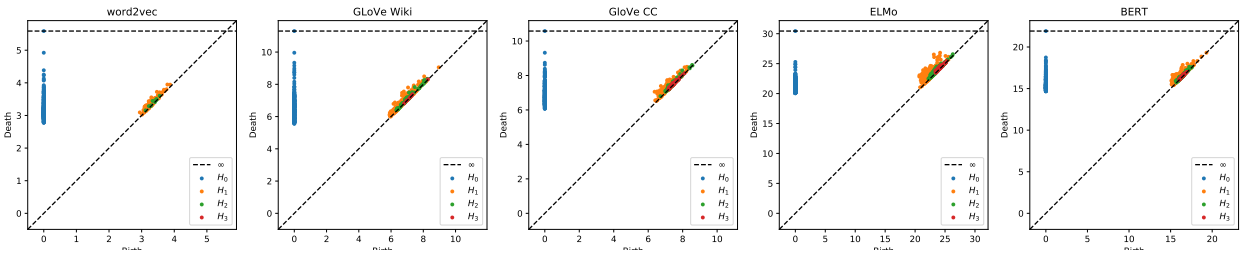
- `poem_2:`



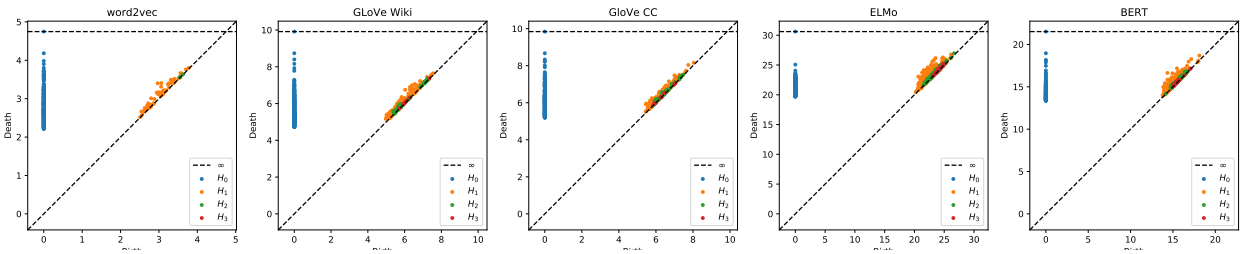
- `poem_3:`



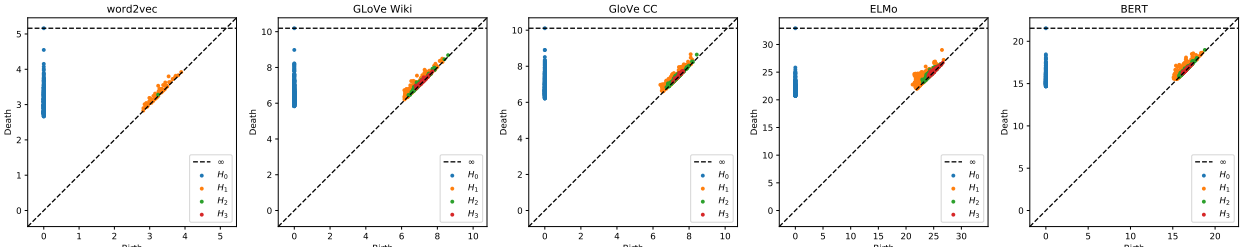
- `press_1:`



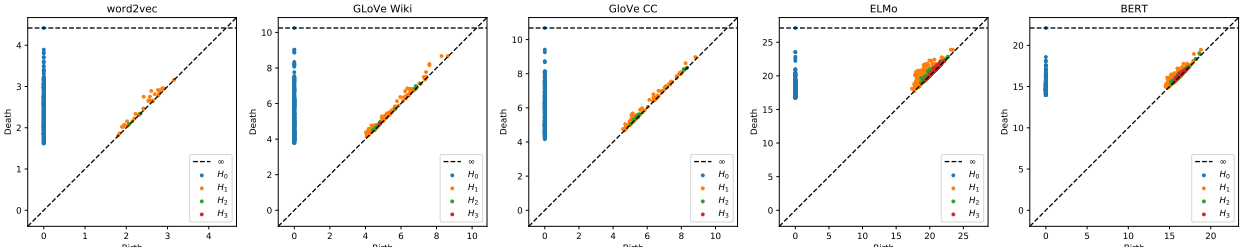
- `press_2:`



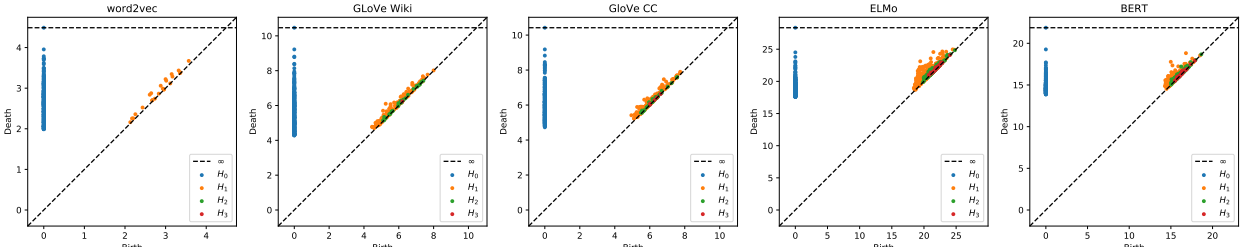
- press_3:



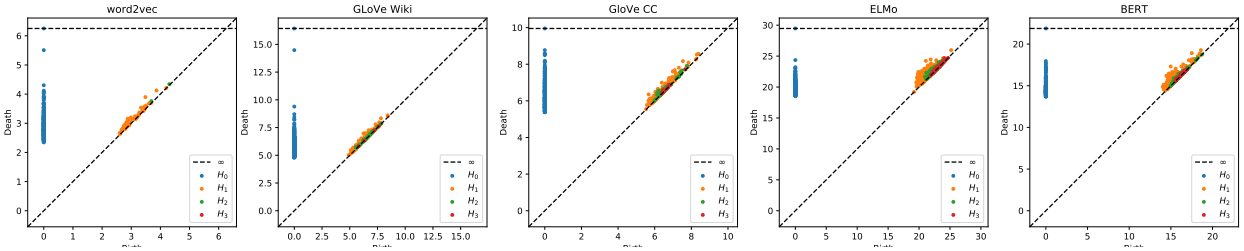
- science_1:



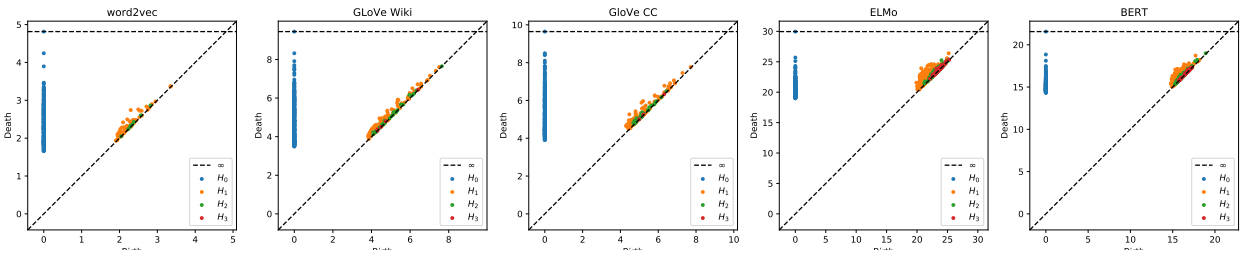
- science_2:



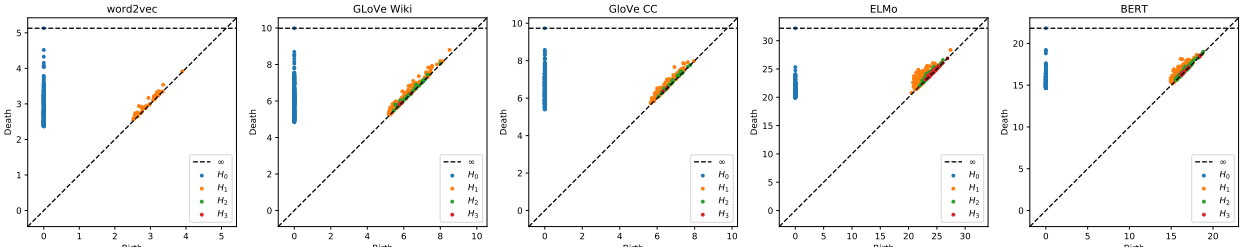
- science_3:



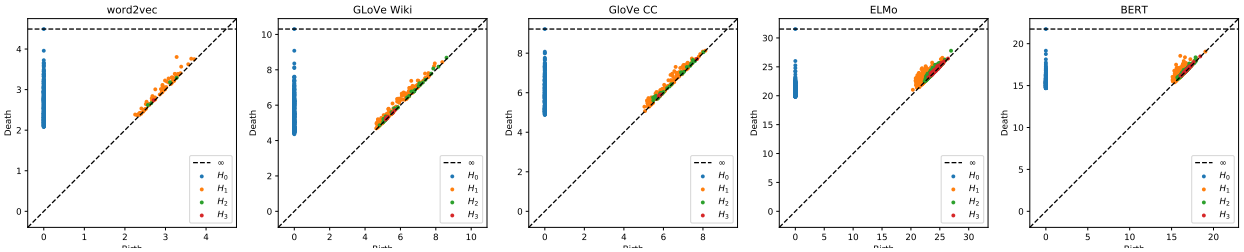
- prose_1:



- prose_2:

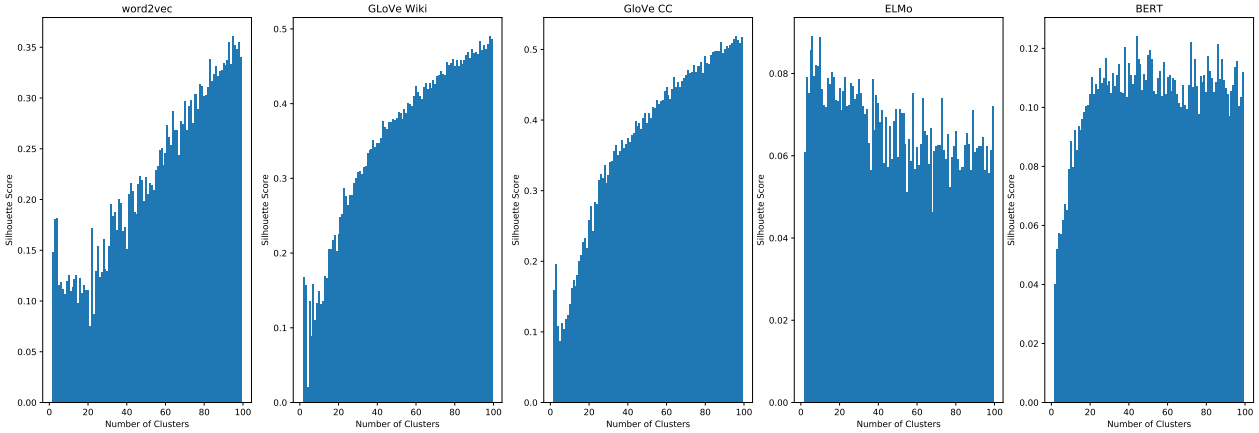


- prose_3:

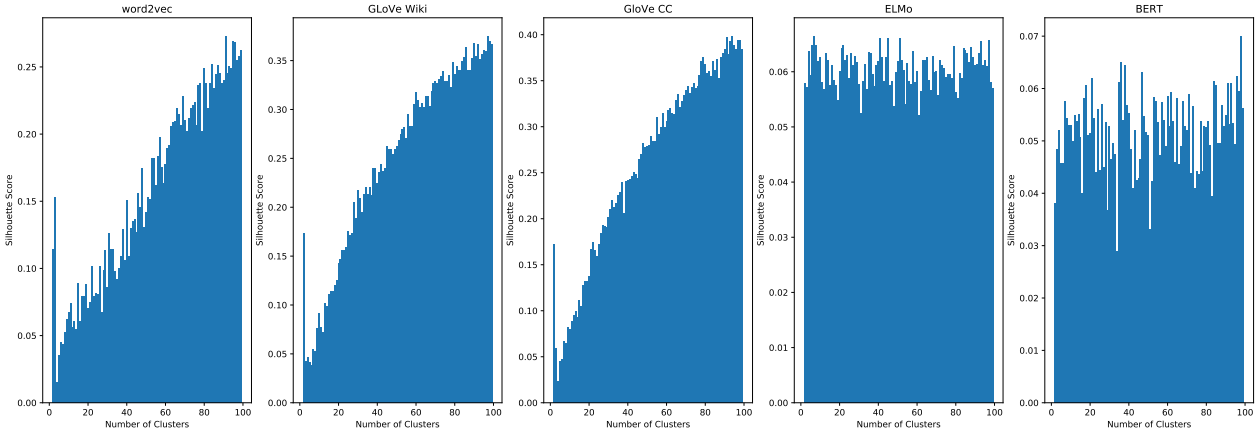


F Other Silhouette Score Plots

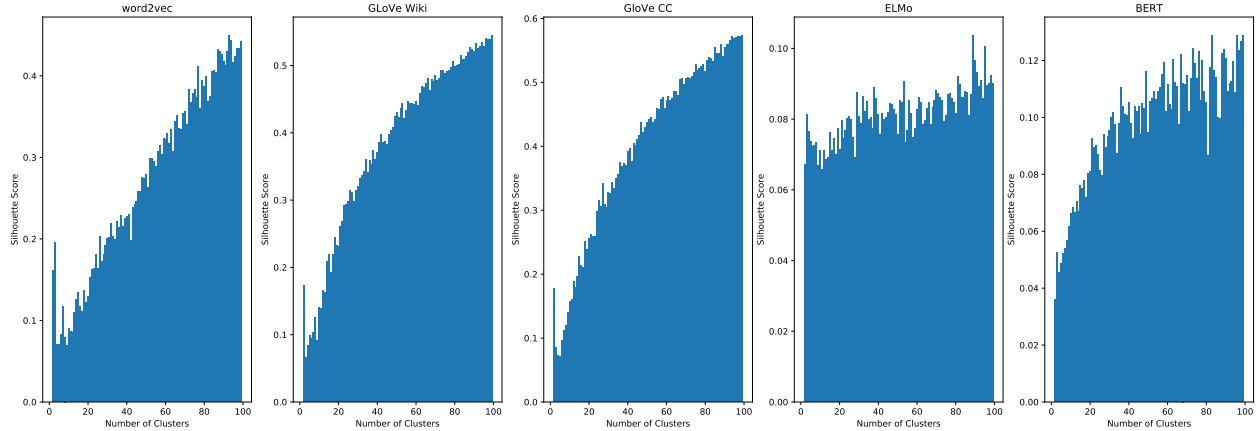
- `poem_1:`



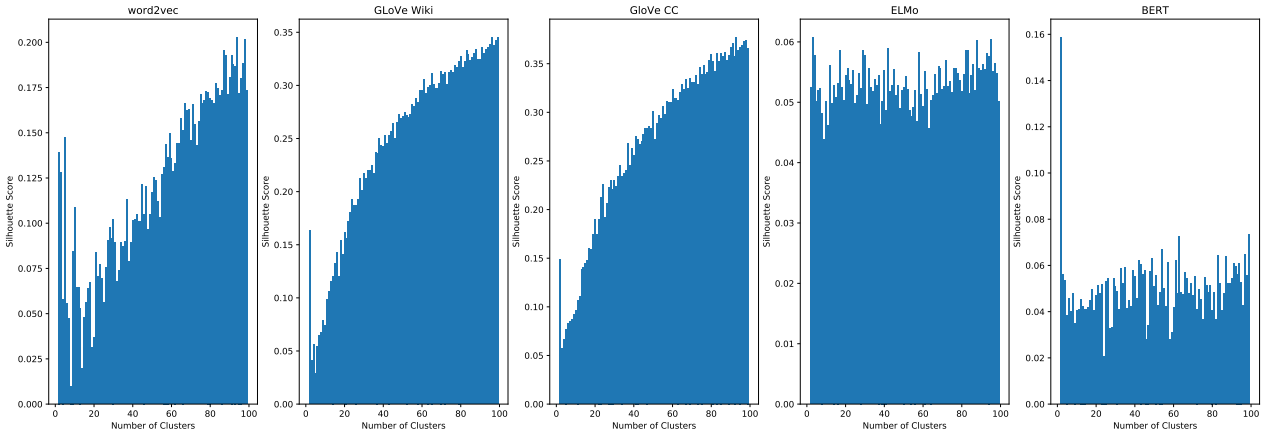
- `poem_2:`



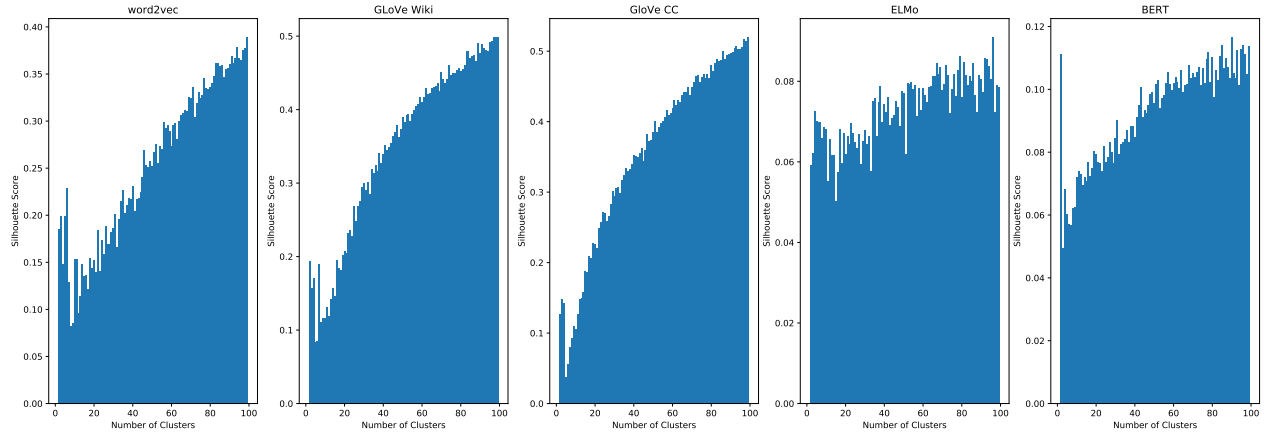
- `poem_3:`



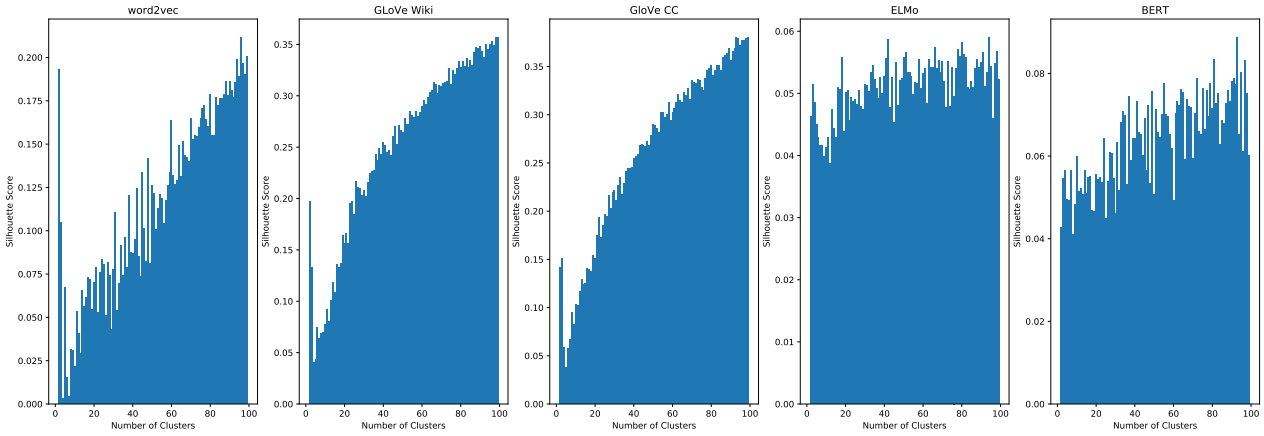
- press_1:



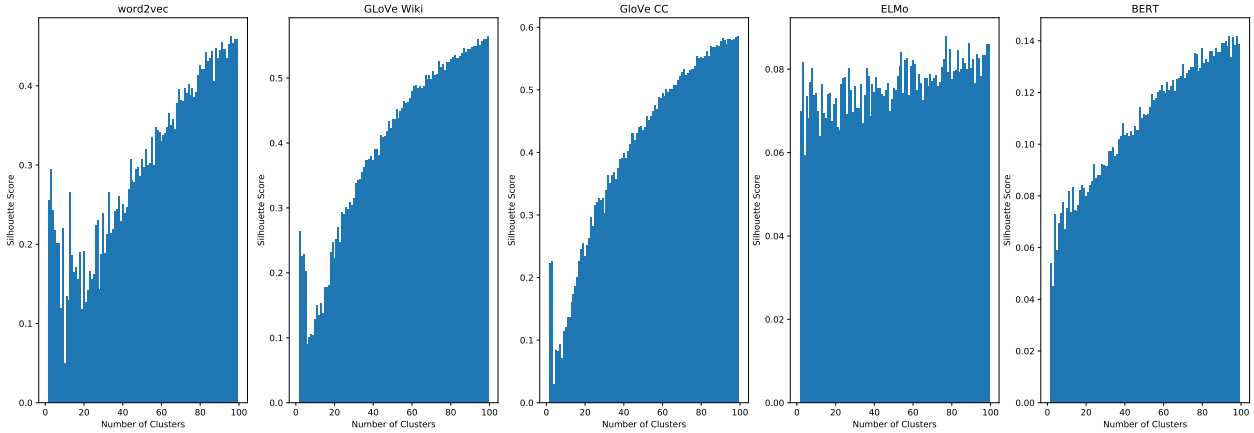
- press_2:



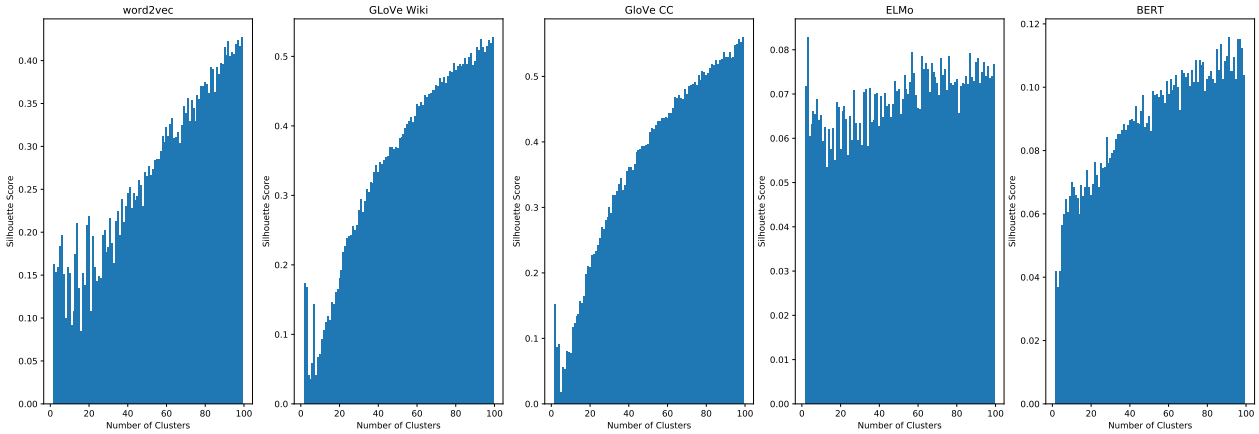
- press_3:



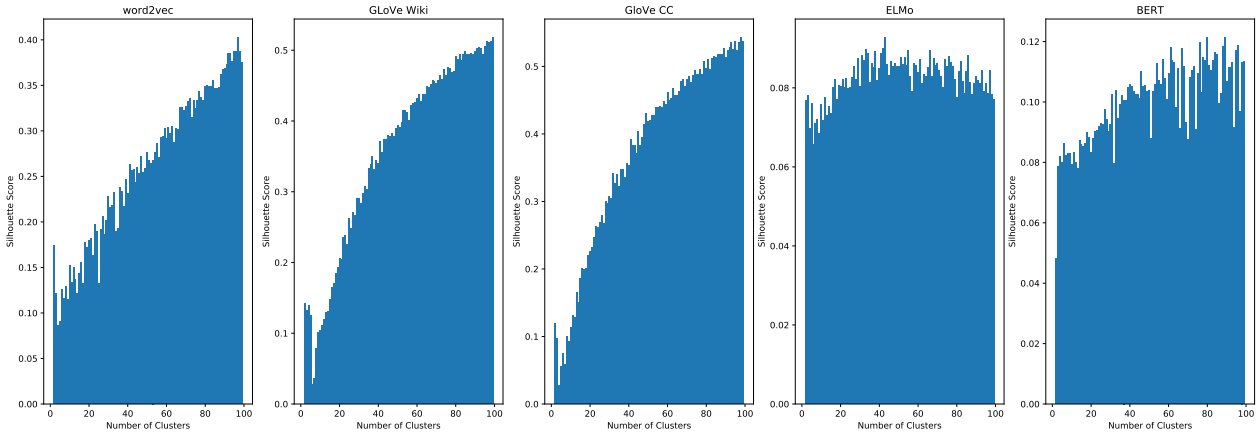
- science_1:



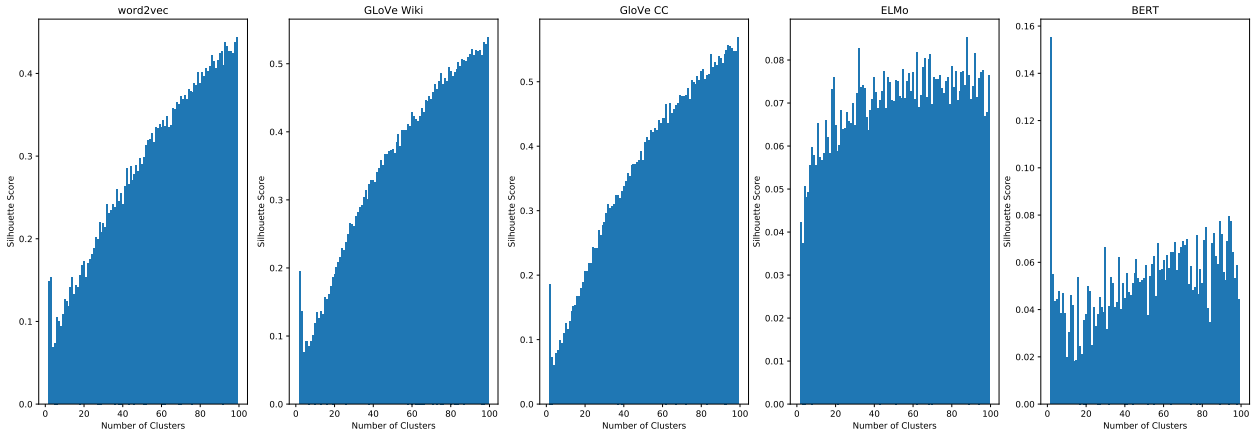
- science_2:



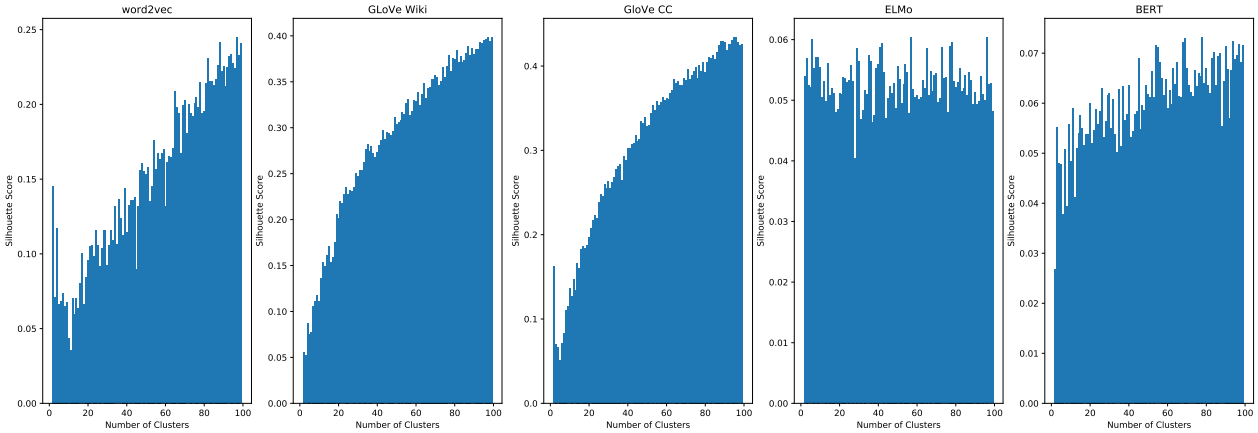
- science_3:



- `prose_1:`



- `prose_2:`



- `prose_3:`