A Tutorial on

Topological Data Analysis

in

Text Mining

Shafie Gholizadeh ¹  Wlodek Zadrozny ¹ ²

¹College of Computing, University of North Carolina at Charlotte

²School of Data Science, University of North Carolina at Charlotte
# Agenda

<table>
<thead>
<tr>
<th>#</th>
<th>Segment</th>
<th>Presentation</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td>Slides</td>
<td>15 min</td>
</tr>
<tr>
<td>2</td>
<td>TDA background &amp; available software</td>
<td>Slides/Code</td>
<td>20 min</td>
</tr>
<tr>
<td>3</td>
<td>Positioning TDA in Text Mining</td>
<td>Slides</td>
<td>15 min</td>
</tr>
<tr>
<td>4</td>
<td>TDA methods for Text Mining</td>
<td>Slides/Code</td>
<td>40 min</td>
</tr>
<tr>
<td>5</td>
<td>Limitations, opportunities &amp; conclusion</td>
<td>Slides</td>
<td>15 min</td>
</tr>
</tbody>
</table>
How do we classify, cluster, or generally analyze text?

- Term Frequency space
- Embeddings
- NNs

Some basic questions:

How do we classify long text?

Why should we limit ourselves to the conventional features?

Is there any alternative to NNs?

This tutorial targets new text representations using topological data analysis.
Introduction: Topological Data Analysis (TDA)

TDA is becoming more popular as a research area.

Publications on TDA

Total Publications: 7,011
Sum of Times Cited: 167,775

---

1Web of Science portal, retrieved on April 20, 2020
Introduction: Topological Data Analysis (TDA)

TDA is becoming more popular as a research area.

Publications on TDA\textsuperscript{2}

\begin{itemize}
    \item 627 CHEMISTRY MULTI-DISCIPLINARY
    \item 591 ENGINEERING ELECTRICAL ELECTRONIC
    \item 566 COMPUTER SCIENCE INTERDISCIPLINARY APPLICATIONS
    \item 478 BIOCHEMISTRY MOLECULAR BIOLOGY
    \item 478 COMPUTER SCIENCE ARTIFICIAL INTELLIGENCE
    \item 470 COMPUTER SCIENCE INFORMATION SYSTEMS
    \item 433 CHEMISTRY PHYSICAL
    \item 412 COMPUTER SCIENCE THEORY METHODS
    \item 405 MATHEMATICAL COMPUTATIONAL BIOLOGY
    \item 313 NEURO-SCIENCES
\end{itemize}

\textsuperscript{2}Web of Science portal, retrieved on April 20, 2020
Some Recent Contributions of TDA

- Clustering
- Dimensionality Reduction
- Descriptive modeling

- Neural Nets Architecture
  
  [Guss and Salakhutdinov, 2018] [Hofer et al., 2019] [Naitzat et al., 2020]

- Sensor Network Coverage
  
  [De Silva and Ghrist, 2006] [Adams and Carlsson, 2015], [Das and DebBarma, 2018]

- Time Series Analysis
- Signal Processing
- Dynamical Systems Analysis
Introduction: Topological Data Analysis (TDA)

How to find shapes in text?

How to use TDA in text processing?
What Is TDA?

- TDA: A collection of methods that find structure of shapes in data.

What’s the Ancestry of TDA?

- Computational Geometry ⇒ Computational Topology ⇒ Topological Data Data Analysis

Common Approach in TDA is to:

1. Capture the shapes as the main characteristics.
2. Dismiss the rest as noise or irrelevant information.
Why Important?
- Use Topological features in addition to the other features.
- Capture the order in the text.

Borrowing Ideas from Time-Series Analysis
- Consider text as $n$-dimensional time-series of $n$ entities.

Designing Order-Preserving Text Processing
Our Contributions

- Providing a New Framework and Algorithms to Extract “Topological Features” from Text:
  - Extracting TDA features from word embeddings space
  - Extracting TDA features from TF-IDF space
  - Extracting TDA features without using conventional representations

- Showing the value of TDA in Text Mining
  - TDA features carrying exclusive information that is not reflected in conventional features.
Betti Numbers Capture Topological Structure.

- The $i^{th}$ Betti number: number of $i$-dimensional holes $a$ in a shape.

<table>
<thead>
<tr>
<th></th>
<th>$\beta_0$</th>
<th>$\beta_1$</th>
<th>$\beta_2$</th>
<th>$\beta_3$</th>
<th>$\beta_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_0$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>\ldots</td>
<td>\ldots</td>
<td>\ldots</td>
<td>\ldots</td>
<td>\ldots</td>
<td>\ldots</td>
</tr>
</tbody>
</table>

- $\beta_0$: Number of connected components
- $\beta_1$: Number of 1-D holes
- $\beta_2$: Number of 2-D voids
Betti Numbers Capture Topological Structure.

- The $i^{th}$ Betti number: number of $i$-dimensional holes $a$ in a shape.

<table>
<thead>
<tr>
<th>$\beta_0$</th>
<th>$\beta_1$</th>
<th>$\beta_2$</th>
<th>$\beta_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_0$</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- $\beta$s are robust under stretching or shrinking.
- $\beta$s simplify complex information.
- Homology studies $\beta$s.
- To find the structure of shapes:
  - Capture the shapes as the main characteristics.
  - Dismiss the rest as noise or irrelevant information.
What Is Persistent Homology?

- High dimensional data sets are:
  “huge number of discrete points”
  ⇒ There are no continuous shapes!
  ⇒ How to define/compute $\beta$’s?
What Is Persistent Homology?

- High dimensional data sets are: “huge number of discrete points”
  ⇒ There are no continuous shapes!
  ⇒ How to define/compute $\beta$’s?

- How visual interpretation works?
  infer a continuous shape from discrete points.

- Translate points into:
  a meaningful topological structure.

---

3[Edelsbrunner et al., 2000, Zomorodian and Carlsson, 2005]
What Is Persistent Homology?

- High dimensional data sets are: “huge number of discrete points”
  ⇒ There are no continuous shapes!
  ⇒ How to define/compute $\beta$’s?

- How visual interpretation works?
  infer a continuous shape from discrete points.

- Translate points into:
  a meaningful topological structure.

- We need **Persistent Homology**$^3$.

---

$^3$[Edelsbrunner et al., 2000, Zomorodian and Carlsson, 2005]
What Is Persistent Homology?

- Decreasing resolution ⇒ Data points get closer to each other.

- Any \(k\) points that get close enough ⇒ Connect them.

- Increasing radius gradually ⇒ Components and Holes (Loops) appear and disappear.
Persistence Diagram Captures Birth and Death Diameters of Holes.

Persistence Diagram

- Radius = 0.01
- Radius = 0.02
- Radius = 0.04
- Radius = 0.05
- Radius = 0.08
- Radius = 0.13
- Radius = 0.18
- Radius = 0.32

Graph showing points labeled as 'Component' and 'Loop'.
Persistence Diagram Captures Birth and Death Diameters of Holes.

Radius = 0.01

Radius = 0.02

Radius = 0.04

Radius = 0.05

Radius = 0.08

Radius = 0.13

Radius = 0.18

Radius = 0.32
Persistence Diagrams Vs. Barcodes

- Persistence diagram: Birth and death of holes shown in 2 dimensions.
- Barcodes [Collins et al., 2004][Ghrist, 2008]: Birth and death of holes shown in 1 dimension.

We will look at the numerical value of barcodes in our work.
Persistence Diagrams Vs. Barcodes

- Persistence diagram: Birth and death of holes shown in 2 dimensions.
- Barcodes [Collins et al., 2004][Ghrist, 2008]: Birth and death of holes shown in 1 dimension.

We will look at the numerical value of barcodes in our work.
Positioning TDA in Text Mining

- TDA vs. TF/IDF feature space
- TDA vs. NN
- TDA vs. Word embeddings
- TDA vs. Transformers
Why Important?
- Using topological features in addition to the other features
- Capturing the Order in the text

Borrowing ideas from Time-Series Analysis
- Considering Text as $n$-dimensional time-series of $n$ entities
- Where vector space representations fail?

Designing Order-Preserving Text Processing
TDA Has Been Applied to Time Series & Signal Processing

- Analysis of Periodic/Quasi-periodic/Recurrent Systems
  - Used Time Delay Embedding. [Skraba et al., 2012]

  ![Time Delay Embedding](image)

- Persistence-Based Clustering on delay embedding [Chazal et al., 2013]

  ![Persistence-Based Clustering](image)

- Step Detection in Periodic Signals [Khasawneh and Munch, 2018]

TDA in Text Mining

- Analyzing Discrepancy on Corpus [Wagner et al., 2012]
  - Using vector space representation of corpus.
  - Make Persistence Diagram using Cosine distances.
  - Diagram is a measure of discrepancy on corpus.

- Finding Signs of “Tie-back” in Documents [Zhu, 2013]
  - Divide document to a fixed number of blocks.
  - Apply persistent homology on TF-IDF space of different blocks.

<table>
<thead>
<tr>
<th></th>
<th>Child Writing</th>
<th>Adolescent Writing</th>
<th>Adolescent (truncated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Holes Existence</td>
<td>87%</td>
<td>100%</td>
<td>98%</td>
</tr>
<tr>
<td>Total Holes</td>
<td>3.0 ± 0.2</td>
<td>17.6 ± 0.9</td>
<td>3.9 ± 0.2</td>
</tr>
</tbody>
</table>


- [Savle et al., 2019] used TDA on term frequency space for text entailment problem.
1 Introduction

2 TDA background and available software

3 Positioning TDA in Text Mining

4 TDA Methods for Text Mining

5 Limitations, opportunities & conclusion
Why Important?
- Using topological features in addition to the other features
- Capturing the Order in the text

Borrowing ideas from Time-Series Analysis
- Considering Text as $n$-dimensional time-series of $n$ entities

Designing Order-Preserving Text Processing
We will use three methods to extract topological features from text.

1. Topological features without using conventional representations
2. Topological features from word embeddings space
3. Topological features from TF/IDF space
(1) Topological Signature of 19th Century Novelists

- An application of Persistent Homology in Text Mining
- 75 Novels form Gutenberg.org by 6 novelists
- Predict author solely based on graph of main characters (=persons).
- Average accuracy in binary classification: 77%
- For each Novel:
  - Find positions of each character (person) in the novel.
    - Use Stanford CoreNLP APIs → named entity recognizer (NER)
    - Find entities tagged as PERSON.
    - Save place that they appeared (Token Indices).
  - Use only 10 most frequent (important?) characters (persons).
  - Measure the distance between character A and character B.

- Using Persistence Diagrams
Topological Signature of 19th Century Novelists

Fyodor Dostoyevsky
The Brothers Karamazov

Fyodor Dostoyevsky
The Idiot

Fyodor Dostoyevsky
The Possessed

Jane Austen
Emma

Jane Austen
Mansfield Park

Jane Austen
Sense and Sensibility

Walter Scott
Guy Mannering

Walter Scott
Kenilworth

Walter Scott
Rob Roy
Topological Signature of 19th Century Novelists

S. Gholizadeh & W. Zadrozny

A Tutorial on TDA in Text Mining

December 2020
Topological Signature of 19th Century Novelists

- Predicting the author
- Binary Classification (balanced sub-samples)
- 250 times 10-fold cross validation
- 60’000 total predictions
- Using a 5-NN algorithm
- Using Wasserstein distance of persistence diagrams

Evaluations (Accuracy)

<table>
<thead>
<tr>
<th></th>
<th>Dickens (17)</th>
<th>Zola (18)</th>
<th>Dostoyevsky (8)</th>
<th>Austen (6)</th>
<th>Twain (8)</th>
<th>Scott (18)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. Dickens</td>
<td>-</td>
<td>87.0</td>
<td>72.2</td>
<td>100.0</td>
<td>74.6</td>
<td>73.9</td>
</tr>
<tr>
<td>É. Zola</td>
<td>87.0</td>
<td>-</td>
<td>65.0</td>
<td>64.2</td>
<td>68.8</td>
<td>83.3</td>
</tr>
<tr>
<td>Dostoyevsky</td>
<td>72.2</td>
<td>65.0</td>
<td>-</td>
<td>90.2</td>
<td>73.3</td>
<td>55.8</td>
</tr>
<tr>
<td>J. Austen</td>
<td>100.0</td>
<td>64.2</td>
<td>90.2</td>
<td>-</td>
<td>82.9</td>
<td>94.7</td>
</tr>
<tr>
<td>M. Twain</td>
<td>74.6</td>
<td>68.8</td>
<td>73.3</td>
<td>82.9</td>
<td>-</td>
<td>68.5</td>
</tr>
<tr>
<td>W. Scott</td>
<td>73.9</td>
<td>83.3</td>
<td>55.8</td>
<td>94.7</td>
<td>68.5</td>
<td>-</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>81.5</strong></td>
<td><strong>73.7</strong></td>
<td><strong>71.3</strong></td>
<td><strong>86.4</strong></td>
<td><strong>73.6</strong></td>
<td><strong>75.2</strong></td>
</tr>
</tbody>
</table>
(2) TDA Using Word Embeddings Space

1. TDA without using conventional features
2. TDA using word embeddings
3. TDA using TF-IDF

- In textual documents:
  - Words are discrete. Similar words are not distinguished.
  - “What if we use word embeddings instead of words”?

- Using word embeddings
  - Try pre-trained fastText, Glove, and ConceptNet Numberbatch.

- Using D-dimensional word embedding
  - Text $\rightarrow$ D-dimensional Time Series

- We study the Topology of D-dimensional Time Series of each text.
Using word embedding with $D = 300$ dimensions,

→ a document with $T$ words: $< \text{Word}_1, \text{Word}_2, \cdots, \text{Word}_T >$

→ can be represented by:

$$\begin{pmatrix}
\text{in} & 0.122 & 0.156 & 0.046 & \cdots & -0.034 \\
\text{the} & 0.124 & 0.167 & 0.033 & \cdots & -0.026 \\
\text{beginning} & 0.118 & 0.082 & 0.009 & \cdots & 0.010 \\
\text{god} & 0.053 & 0.040 & -0.016 & \cdots & 0.134 \\
\text{created} & 0.110 & 0.035 & -0.003 & \cdots & -0.029 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\text{coffin} & 0.035 & 0.019 & 0.110 & \cdots & 0.025 \\
\text{in} & 0.122 & 0.156 & 0.046 & \cdots & -0.034 \\
\text{egypt} & -0.094 & 0.043 & 0.014 & \cdots & -0.013
\end{pmatrix}$$

$\text{Genesis}_{T \times 300}$
A Novel Algorithm for Document Representation

- A textual Doc: A D-dimensional time series (input)

- Define some distance among D embedding dimensions.
  - Based on Cosine Similarity, Correlation, or Covariance

- For each document:
  - ⇒ Make a graph of D vertices.
  - ⇒ Get the persistent diagram of the graph.
  - ⇒ How much the PD will change if we exclude dimension d?
    - ⇒ Do it for $d = 1, \ldots, D$.
  - ⇒ “Differentiate” the PD’s using Wasserstein distance.
  - ⇒ Get D topological features (output) for each document.
What Is the Algorithm Doing?

- For each document:
  - ⇒ Build persistent diagram.
  - ⇒ Get the sensitivity of PD to each embedding dimension.
  - ⇒ Use it as the sensitivity of the document itself to that embedding dimension.

- Use these features for classification, clustering, etc.

- Is the final result a “Document Embedding”? 

S. Gholizadeh & W. Zadrozny
A Tutorial on TDA in Text Mining
December 2020 28
Why Differentiating Persistent Diagram?

Coordinates (Graph) of embedding dimensions

Graph as PH perceives

Persistent homology does not distinguish the order of dimensions if we do not differentiate the results.
### Toward Order Preserving Algorithm

- We define the distance between embedding dimensions based on Covariance/Correlation/Cosine Matrix → It is **not order preserving**.
  - Like a bag-of-words model, shuffling the words produces same results.

- What if we **smooth the time-series** first? After smoothing, each index of an embedding dimension is being compared to:
  1. the same index of other dimensions
  2. a few lags/leads of other dimensions

![Time-series Matrix](image)

### Mathematical Formulation

Let $X_{1}(t)$, $X_{d}(t)$, and $X_{D}(t)$ represent embedding dimensions for different time points $t$. The matrix $X(t)$ can be represented as:

$$X(t) = \begin{bmatrix} X_{1}(1) & \ldots & X_{d}(1) & \ldots & X_{D}(1) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{1}(t) & \ldots & X_{d}(t) & \ldots & X_{D}(t) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{1}(T) & \ldots & X_{d}(T) & \ldots & X_{D}(T) \end{bmatrix}_{T \times D}$$
**arXiv Papers**: We downloaded all of arXiv papers in quantitative finance\(^4\) published between 2011 and 2018.

- We selected five major categories (subject tags): General Finance, Statistical Finance, Mathematical Finance, Pricing of Securities, and Risk Management.
- We topological features to XGBoost to classify arXiv topics.
- Achieved F1-score of 0.643.
  -> Outperforming our best CNN classifier where F1-score was 0.607.

**IMDB Movie Review** [Maas et al., 2011]: Using IMDB reviews annotated by positive/negative labels, we examined the topological algorithm on word embeddings for binary sentiment classification.

- Fed topological features to XGBoost to classify movie review polarity:
  - Achieved F1-score of 0.884.
  -> Slightly better than the best previous results where F1 was 0.880.

\(^4\)https://arXiv.org/archive/q-fin
(3) TDA Using TF-IDF Space

1. TDA without using conventional features
2. TDA using word embeddings
3. TDA using TF-IDF

Similar to the [Zhu, 2013] method, the idea is to:

(a) Dividing the textual document to a fixed number of blocks.

(b) Analyze the cosine distances among different blocks.

(c) Search for repetitive patterns among the blocks.

The results might be not as strong as TDA features from WE. But we may use them in addition to other features for text classification.
TDA Using TF-IDF: Details of the Algorithm

For each document:

1. Divide it to 10 blocks.
2. Calculate the TF-IDF of each block.
3. Calculate the Cosine similarity among different blocks.
   \[\Rightarrow\] We have a weighted graph whose vertices are 10 text blocks.
4. Apply persistent homology on the graph.
   \[\Rightarrow\] We will get birth/death diameters for dimension 0 (components).
   \[\Rightarrow\] We will get birth/death diameters for dimension 1 (loops)
TDA Using TF-IDF: Details of the Algorithm

Threshold = 0.10

Threshold = 0.21

Threshold = 0.30

Threshold = 0.40

Persistence Diagram

![Persistence Diagram with Component and Loop markers](image)
For each document:

- In dimension 0 (components):
  - All the birth diameters are always zero.
  - We always get 9 diameter of death. \( \Rightarrow \) We get 9 features.

- In dimension 1 (loops):
  - We may see different number of loops for different documents.
  - How to flatten the results? A trick: Retrieve only 5 statistics.
    1. number of loops
    2. average diameter of birth
    3. average diameter of duration (duration = death - birth)
    4. standard deviation of birth diameters
    5. standard deviation of duration diameters

- Totally, we get 9 + 5 = 14 features for the document.
Evaluation of TDA on TF-IDF & Word Embeddings

- We run both algorithms for (1) word embeddings and (2) TF-IDF on Wikipedia Movie Plots from Kaggle. Selected four major genres.

- Fed the topological features to XGBoost to predict the genres. Also tried BiLSTM to benchmark the results.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Pre.</th>
<th>Rec.</th>
<th>F1</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 BiLSTM</td>
<td>68.0</td>
<td>59.7</td>
<td>0.608</td>
<td>76.2</td>
</tr>
<tr>
<td>2 XGBoost on TP1</td>
<td>59.6</td>
<td>53.2</td>
<td>0.560</td>
<td>71.1</td>
</tr>
<tr>
<td>3 XGBoost on TP1 &amp; TP2</td>
<td>59.9</td>
<td>53.7</td>
<td>0.564</td>
<td>71.4</td>
</tr>
<tr>
<td>4 BiLSTM + XGBoost on TP1</td>
<td>67.8</td>
<td>64.8</td>
<td>0.656</td>
<td>77.3</td>
</tr>
<tr>
<td>5 BiLSTM + XGBoost on TP1 &amp; TP2</td>
<td>68.5</td>
<td>64.6</td>
<td><strong>0.659</strong></td>
<td><strong>77.8</strong></td>
</tr>
</tbody>
</table>

- TP1: Topological features from Word embeddings.
- TP2: Topological features from TF-IDF.

*Topological features from word embeddings are much more helpful than topological features from TF-IDF.*
Contents

1. Introduction

2. TDA background and available software

3. Positioning TDA in Text Mining

4. TDA Methods for Text Mining

5. Limitations, opportunities & conclusion
Limitations ≈ Opportunities

- Only classification here, and some *inference* ([Savle et al., 2019])
- What about *summarization*?
  - Embeddings represent semantic/conceptual connections, and topology compresses. Could this be helpful for summarization?
- What about other data sets?
- Newest *attention models* not investigated
  - In principle NN can find topological features of any data
- Mapping back from topology to text:
  - Poorly understood reasons for improvements
  - Simplexes as a representation of concept boundaries?
  - Principle-based compression?
We introduced and evaluated three methods of extracting topological representations from text:

1. using TF-IDF vector space,
2. using word embeddings space, and
3. using name entities without any conventional features.

Topological representation extracted without using conventional features is primarily useful for author profiling/classification.

Features extracted from word embeddings showed the best performance, while features from TF-IDF are primarily designated to find repetitive patterns in text.

TDA features carrying exclusive information that is not reflected in conventional features. They can boost the classification results.
Future Direction

- Analyze the co-appearances of more entity types. We analyzed only the name entities tagged as ‘person’. We can extend it to locations, POS tags, etc.

- We mainly focused on text classification. We will investigate of extend and apply our methods for other natural language processing tasks, such as summarization or question answering.

- Apply persistent homology on attention models (the attention matrices).

- A open problem, explainability: Find the actual text behind the topological structures.
Questions?


Movie genre detection using topological data analysis.

Topological persistence and simplification.

Barcodes: the persistent topology of data.

Sentiment analysis leveraging emotions and word embeddings.

Topological data analysis of critical transitions in financial networks.

Topological data analysis of financial time series: Landscapes of crashes.

On characterizing the capacity of neural networks using algebraic topology.
Bibliography III

Connectivity-optimized representation learning via persistent homology.
In International Conference on Machine Learning, pages 2751–2760.

Bag of tricks for efficient text classification.

Topological data analysis for true step detection in periodic piecewise constant signals.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011).
Learning word vectors for sentiment analysis.

Topological characterization and early detection of bifurcations and chaos in complex systems using persistent homology.


Glove: Global vectors for word representation.
Topological strata of weighted complex networks.

Topological data analysis for discourse semantics?
In *Proceedings of the 13th International Conference on Computational Semantics-Student Papers*, pages 34–43.

Sentiment analysis on imdb using lexicon and neural networks.

Topological analysis of recurrent systems.

Conceptnet 5.5: An open multilingual graph of general knowledge.
In *Thirty-First AAAI Conference on Artificial Intelligence*.

Computational topology in text mining.

Persistent homology: An introduction and a new text representation for natural language processing.
Computing persistent homology.
*Discrete & Computational Geometry*, 33(2):249–274.
Persistent Landscape [Bubenik, 2015]: Real-valued function

Intuitively rotate persistence diagram by $\pi/4$

$$\lambda: \mathbb{N} \times \mathbb{R} \rightarrow \mathbb{R}$$

$$\lambda(n, t) = \sup \{\text{radius} \geq 0 \mid \beta(t - \text{radius}, t + \text{radius}) \geq n\}$$

$$\forall n \in \mathbb{N}$$
**Rips Filtration** [Ghrist, 2008]: A $k$-simplex has $k$ nodes with pairwise distance $\leq \epsilon$.

**Čech Complex**: Slightly stronger conditions: The regions around the nodes of a simplex within the radii equal to the $\epsilon$ altogether should have a non-empty intersection.

**Weight Rank Clique Filtration** [Petri et al., 2013]: Dealing with a weighted graph (instead of a data cloud): Threshold the weights and increase the weights threshold gradually.
Clustered Iris data set
(the labels give the true flower species)

Wasserstein Distance: Intuition
Topological Signature of 19th Century Novelists

How to define distances?

Distance of character A and character B

\[ \text{Distance}_t(A, B) = WD_{0.5}(\tilde{I}^{1+t}, \tilde{J}^{1+t}) \]

- \( \tilde{I}, \tilde{J} \): normalized indices of positions where A and B appear respectively
- \( t = 0 \rightarrow \) Wasserstein distance of order 0.5 of \( \tilde{I}, \tilde{J} \)
- Order 0.5 \( \rightarrow \) sensitive to the closer element-wise distances
- Where they co-appear? \( t = 0, -\epsilon, \text{and} + \epsilon \)

Distance of novel X and novel Y

\[ \text{Distance}_t(X, Y) = WD\{PD_t^0(X), PD_t^0(Y)\} + WD\{PD_t^1(X), PD_t^1(Y)\} \]

\[ \text{Distance}(X, Y) = \left\{ \sum_{t \in \{-\epsilon, 0, +\epsilon\}} \text{Distance}_t(X, Y)^2 \right\}^{\frac{1}{2}} \]
Covariance/Correlation/Cosine Matrix is not order preserving. Like a bag-of-words model, shuffling the words produces same results.

What if we use a smoothed time-series?

Using exponential smoothing?
Using local averages (sliding window of size $\omega = 5$):

$$\tilde{X}_i = \tilde{X}_i(t) = X_i(t - 2) + X_i(t - 1) + \cdots + X_i(t + 2)$$
Covariance/Correlation/Cosine Matrix is not order preserving.
Like a bag-of-words model, shuffling the words produces same results.

What if we use a smoothed time-series?

Using exponential smoothing?
Using local averages (sliding window of size $\omega = 5$):

\[
\tilde{X}_i(t) = \tilde{X}_i(t) = X_i(t-2) + X_i(t-1) + \cdots + X_i(t+2)
\]

\[
\mathbb{E}[\tilde{X}_i \tilde{X}_j] = \mathbb{E}[\tilde{X}_i(t) \tilde{X}_j(t)] \\
\approx 5\mathbb{E}[X_i(t)X_j(t)] \\
+ 4\mathbb{E}[X_i(t-1)X_j(t)] + 4\mathbb{E}[X_i(t)X_j(t-1)] \\
+ 3\mathbb{E}[X_i(t-2)X_j(t)] + 3\mathbb{E}[X_i(t)X_j(t-2)] \\
+ 2\mathbb{E}[X_i(t-3)X_j(t)] + 2\mathbb{E}[X_i(t)X_j(t-3)] \\
+ 1\mathbb{E}[X_i(t-4)X_j(t)] + 1\mathbb{E}[X_i(t)X_j(t-4)]
\]
Toward Order Preserving Algorithm

- Covariance/Correlation/Cosine Matrix is not order preserving.
  - Like a bag-of-words model, shuffling the words produces same results.

- What if we use a smoothed time-series?
  - Using exponential smoothing?
  - Using local averages (sliding window of size $\omega = 5$):
    $$\tilde{X}_i = \tilde{X}_i(t) = X_i(t - 2) + X_i(t - 1) + \cdots + X_i(t + 2)$$

\[
E[\tilde{X}_i \tilde{X}_j] = E[\tilde{X}_i(t) \tilde{X}_j(t)] \\
\approx 5E[X_i(t) X_j(t)] \\
\quad + 4E[X_i(t - 1) X_j(t)] + 4E[X_i(t) X_j(t - 1)] \\
\quad + 3E[X_i(t - 2) X_j(t)] + 3E[X_i(t) X_j(t - 2)] \\
\quad + 2E[X_i(t - 3) X_j(t)] + 2E[X_i(t) X_j(t - 3)] \\
\quad + 1E[X_i(t - 4) X_j(t)] + 1E[X_i(t) X_j(t - 4)]
\]
Distance among Embedding Dimensions

- How to Define Distance between Embedding Dimensions?

\[
\varphi(\tilde{X}_i, \tilde{X}_j) := \sqrt{\mathbb{E}[\tilde{X}_i^2] \mathbb{E}[\tilde{X}_j^2] - \mathbb{E}[\tilde{X}_i \tilde{X}_j]}
\]

\[
= \frac{1}{T} \| \tilde{X}_i \| \| \tilde{X}_j \| - \frac{1}{T} \tilde{X}_i^T \tilde{X}_j
\]

\[
= \frac{1}{T} \| \tilde{X}_i \| \| \tilde{X}_j \| \{1 - \text{CosSim}(\tilde{X}_i, \tilde{X}_j)\}
\]

- Desired properties:
  1. Insensitive to the length of document
  2. Sensitive to the magnitude of the signal
  3. Increasing function of cosine distance
We utilize some widely used word embeddings:

- **GloVe**\(^5\) pre-trained on Wikipedia 2014 and Gigaword 5 with vocabulary size of 400K and 300d vectors

- **fastText**\(^6\) pre-trained on Wikipedia 2017 with the vocabulary size of 1M and 300d vectors

- **ConceptNet Numberbatch**\(^7\) \(v17.06\) with the vocabulary size of 400K and 300d vectors

\(^5\) [Pennington et al., 2014]
\(^6\) [Bojanowski et al., 2016, Joulin et al., 2016]
\(^7\) [Speer et al., 2017]
Date Specification for arXiv papers and IMDB reviews.

<table>
<thead>
<tr>
<th>Specification</th>
<th>arXiv Quant. Fin. Papers</th>
<th>IMDB Movie Reviews</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labels</td>
<td>5 (Multi-label)</td>
<td>2</td>
</tr>
<tr>
<td>Clean Records</td>
<td>4601</td>
<td>6000</td>
</tr>
<tr>
<td>Length of Records</td>
<td>8456.9 ± 6395.8</td>
<td>540.5 ± 171.9</td>
</tr>
</tbody>
</table>

 Frequency of Labels

- $q$-fin.GN : 1258
- $q$-fin.ST : 1144
- $q$-fin.MF : 977
- $q$-fin.PR : 907
- $q$-fin.RM : 913

Positive : 3000
Negative : 3000
Histograms of number of labels per document in arXiv data set of papers.
## Results using TDA on Word Embeddings

On **arXiv papers** dataset, TDA achieves better F1 and Acc.

<table>
<thead>
<tr>
<th>Model</th>
<th>Embedding</th>
<th>Window</th>
<th>Prec.</th>
<th>Rec.</th>
<th>F1</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>3</td>
<td>61.9</td>
<td>55.4</td>
<td>0.575</td>
<td>80.1</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>3</td>
<td>63.1</td>
<td>56.7</td>
<td>0.597</td>
<td>80.7</td>
</tr>
<tr>
<td><strong>Topology + XGBoost</strong></td>
<td>Numberbatch</td>
<td>3</td>
<td><strong>68.7</strong></td>
<td><strong>60.5</strong></td>
<td><strong>0.643</strong></td>
<td><strong>82.6</strong></td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>5</td>
<td>60.8</td>
<td>54.7</td>
<td>0.576</td>
<td>79.8</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>5</td>
<td>61.8</td>
<td>56.1</td>
<td>0.588</td>
<td>80.3</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>5</td>
<td>65.5</td>
<td>58.4</td>
<td>0.617</td>
<td>81.6</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>7</td>
<td>58.9</td>
<td>54.4</td>
<td>0.566</td>
<td>79.5</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>7</td>
<td>62.8</td>
<td>56.4</td>
<td>0.594</td>
<td>80.6</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>7</td>
<td>65.7</td>
<td>57.7</td>
<td>0.614</td>
<td>81.3</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>7 expon.</td>
<td>60.3</td>
<td>54.6</td>
<td>0.573</td>
<td>79.7</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>7 expon.</td>
<td>61.2</td>
<td>55.9</td>
<td>0.584</td>
<td>80.2</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>7 expon.</td>
<td>66.4</td>
<td>59.6</td>
<td>0.628</td>
<td>82.2</td>
</tr>
<tr>
<td>CNN</td>
<td>fastText</td>
<td>-</td>
<td>57.1</td>
<td>64.3</td>
<td>0.605</td>
<td>80.0</td>
</tr>
<tr>
<td>CNN</td>
<td>GloVe</td>
<td>-</td>
<td>57.6</td>
<td>64.2</td>
<td>0.607</td>
<td>80.6</td>
</tr>
<tr>
<td>CNN</td>
<td>Numberbatch</td>
<td>-</td>
<td><strong>55.0</strong></td>
<td><strong>67.6</strong></td>
<td><strong>0.607</strong></td>
<td><strong>79.8</strong></td>
</tr>
</tbody>
</table>
Results using TDA on Word Embeddings

On **IMDB Reviews** dataset, TDA achieves slightly better F1 and Acc.

<table>
<thead>
<tr>
<th>Model</th>
<th>Embedding</th>
<th>Window</th>
<th>Prec.</th>
<th>Rec.</th>
<th>F1</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>3</td>
<td>84.8</td>
<td>85.8</td>
<td>0.853</td>
<td>85.4</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>3</td>
<td>86.9</td>
<td>88.0</td>
<td>0.874</td>
<td>87.5</td>
</tr>
<tr>
<td><strong>Topology + XGBoost</strong></td>
<td>Numberbatch</td>
<td>3</td>
<td><strong>87.9</strong></td>
<td><strong>89.0</strong></td>
<td><strong>0.884</strong></td>
<td><strong>88.5</strong></td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>5</td>
<td>84.2</td>
<td>85.2</td>
<td>0.847</td>
<td>84.8</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>5</td>
<td>85.6</td>
<td>86.6</td>
<td>0.861</td>
<td>86.2</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>5</td>
<td>86.5</td>
<td>87.6</td>
<td>0.870</td>
<td>87.1</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>7</td>
<td>82.8</td>
<td>83.8</td>
<td>0.833</td>
<td>83.4</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>7</td>
<td>83.8</td>
<td>84.8</td>
<td>0.843</td>
<td>84.4</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>7</td>
<td>85.3</td>
<td>86.3</td>
<td>0.858</td>
<td>85.9</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>fastText</td>
<td>7 expon.</td>
<td>84.3</td>
<td>85.3</td>
<td>0.848</td>
<td>84.9</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>GloVe</td>
<td>7 expon.</td>
<td>86.5</td>
<td>87.6</td>
<td>0.870</td>
<td>87.1</td>
</tr>
<tr>
<td>Topology + XGBoost</td>
<td>Numberbatch</td>
<td>7 expon.</td>
<td>87.0</td>
<td>88.1</td>
<td>0.875</td>
<td>87.6</td>
</tr>
<tr>
<td>[Shaukat et al., 2020]</td>
<td>Lexicon based</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td>86.7</td>
</tr>
<tr>
<td>[Giatsoglou et al., 2017]</td>
<td>Hybrid (TF/IDF</td>
<td>-</td>
<td></td>
<td></td>
<td><strong>0.880</strong></td>
<td><strong>87.8</strong></td>
</tr>
<tr>
<td></td>
<td>+ Embeddings</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results using TDA on Word Embeddings

Results per class on arXiv papers dataset using ConceptNet Numberbatch as pre-trained embedding and window size of 3.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Test Records</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>q-fin.GN</td>
<td>410</td>
<td>73.2</td>
<td>68.5</td>
<td>0.708</td>
<td>83.8</td>
</tr>
<tr>
<td>q-fin.ST</td>
<td>396</td>
<td>70.2</td>
<td>67.5</td>
<td>0.688</td>
<td>83.6</td>
</tr>
<tr>
<td>q-fin.MF</td>
<td>306</td>
<td>66.0</td>
<td>45.6</td>
<td>0.539</td>
<td>77.5</td>
</tr>
<tr>
<td>q-fin.PR</td>
<td>305</td>
<td>69.5</td>
<td>55.2</td>
<td>0.615</td>
<td>82.7</td>
</tr>
<tr>
<td>q-fin.RM</td>
<td>307</td>
<td>62.5</td>
<td>61.0</td>
<td>0.617</td>
<td>84.5</td>
</tr>
</tbody>
</table>

The best F1 achieved for “General” class, while the worst case is for “Mathematical Finance” probably because the set of its class-specific terms has many intersections with other classes.
### Number of records per class and overlaps

<table>
<thead>
<tr>
<th>Specification</th>
<th>Drama</th>
<th>Comedy</th>
<th>Action</th>
<th>Romance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlap with drama</td>
<td>-</td>
<td>524</td>
<td>223</td>
<td>379</td>
</tr>
<tr>
<td>Overlap with comedy</td>
<td>524</td>
<td>-</td>
<td>207</td>
<td>544</td>
</tr>
<tr>
<td>Overlap with action</td>
<td>223</td>
<td>207</td>
<td>-</td>
<td>117</td>
</tr>
<tr>
<td>Overlap with romance</td>
<td>379</td>
<td>544</td>
<td>117</td>
<td>-</td>
</tr>
<tr>
<td>Exclusive Records</td>
<td>4592</td>
<td>3302</td>
<td>1181</td>
<td>672</td>
</tr>
<tr>
<td>Total Records</td>
<td>5615</td>
<td>4477</td>
<td>1658</td>
<td>1614</td>
</tr>
</tbody>
</table>
Our ensemble models provide the best results for every genre.

<table>
<thead>
<tr>
<th>Class</th>
<th>BiLSTM</th>
<th>XGB</th>
<th>XGB2</th>
<th>LR</th>
<th>LR2</th>
<th>prev.SVC</th>
<th>prev.NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>action</td>
<td>87.7</td>
<td>86.7</td>
<td>86.9</td>
<td>89.3</td>
<td>88.9</td>
<td>81.5</td>
<td>82.7</td>
</tr>
<tr>
<td>comedy</td>
<td>75.6</td>
<td>69.0</td>
<td>69.1</td>
<td>76.9</td>
<td>77.7</td>
<td>74.6</td>
<td>73.3</td>
</tr>
<tr>
<td>drama</td>
<td>69.9</td>
<td>63.9</td>
<td>64.3</td>
<td>71.0</td>
<td>71.6</td>
<td>66.1</td>
<td>67.4</td>
</tr>
<tr>
<td>romance</td>
<td>87.6</td>
<td>86.0</td>
<td>85.9</td>
<td>87.8</td>
<td>87.8</td>
<td>88.3</td>
<td>84.3</td>
</tr>
<tr>
<td>macro-avg</td>
<td>76.2</td>
<td>71.1</td>
<td>71.4</td>
<td>77.3</td>
<td>77.8</td>
<td>73.5</td>
<td>73.3</td>
</tr>
</tbody>
</table>

- XGB: XGBoost using TP1 (Topological features of word embeddings)
- XGB2: XGBoost using both topological feature sets.
- LR: Logistic regression combining the results of XGB and BiLSTM
- LR2 : Logistic regression combining the results of XGB2 and BiLSTM
- prev SVC: previous results using linear SVC
- prev NB: previous results using multinomial Naïve Bayes.
Results using TDA on TF-IDF & Word Embeddings

- Topological features from word embedding space can classify the records alone with an accuracy comparable but not equal to the LSTM.

- Topological features extracted from TF-IDF space are primarily used to reflect some repetitive patterns in the text.

- Using the topological feature sets can boost the accuracy of classification in the ensemble model.
### Table: Features from dimension 1 extracted from TF-IDF representation.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Loops Info.</th>
<th>Prev. Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Number of loops</td>
<td>Loops variety</td>
<td>[Zhu, 2013] to find repetitive text patterns</td>
</tr>
<tr>
<td>2 Avg diameter of birth</td>
<td>Birth location</td>
<td></td>
</tr>
<tr>
<td>4 Avg duration</td>
<td>Duration location</td>
<td>[Mittal and Gupta, 2017] for dynamical systems</td>
</tr>
<tr>
<td>3 Std Dev of birth diameters</td>
<td>Birth scale</td>
<td></td>
</tr>
<tr>
<td>5 Std Dev of durations</td>
<td>Duration scale</td>
<td></td>
</tr>
</tbody>
</table>
Introduction: Topology vs. Geometry

- Voronoi Diagram: partitioning the space to convex sub-regions.

- Geometry depends on distances.

- Geometry
  ⇒ Voronoi diagram
  ⇒ Distance-based models in machine learning

- TDA techniques utilize:
  - similar intuition
  - more complex methodologies

- Topology is robust under stretching or shrinking.

- TDA methods are much less sensitive to the choice of metric.
Can AI Consider Complicated Scenarios?

A moment of tension in Vatican. If the bishop moves forward the queen can take him.