Topic Discovery, Summarisation, and Diffusion

Roy Gardner

PeaceRep, University of Edinburgh

Contents

1 Introduction

1.1 Definitions

Topic discovery The process of finding clusters of semantically similar sentence–level text segments in document sets. A cluster of semantically similar text segments constitutes a candidate topic. Candidate topics are assessed by domain experts (see Appendix A).

Summarisation The process of selecting one or more summary segments to represent a topic. If a topic spans two or more document sets, then a topic summary contains a segment from each set. Summary segments are used in downstream processing, for example, as templates for topic search, and for designing new topics for use in automatic classification of document text.

Diffusion The tracking of topics across a time series of documents.

1.2 Measuring Semantic Similarity

Sentence-level semantic similarity measures the degree to which two or more natural language sentences or clauses convey similar meaning. We use version 4 of Google's Universal Sentence Encoder (USE v4)[1][2] to generate 512-length numerical representations of sentences referred to as encoding vectors or embeddings.

The preferred measurement of the distance between a pair of encoding vectors is angular distance. The inverse of this distance is a score of the semantic similarity between the sentences that the vectors represent. Semantic similarity scores range from 1.0 to 0.0 where 1.0 means two sentences are identical in meaning and comprise the same words in the same order. As the meaning of the sentences diverge, the similarity score decreases.

USE models enable efficient and accurate computation of sentence-level encoding vectors, making it possible to perform large-scale semantic similarity tasks on a range of multi-language datasets without any pre-processing of text other than segmentation and encoding[3].

[1] https://www.kaggle.com/models/google/universal-sentence-encoder/frameworks/tensorFlow2/variations/universal-sentence-encoder/versions/2?tfhub-redirect=true

2 Methods

The methodology uses graph-based clustering of binary-valued matrices to find clusters of semantically similar text segments within or across document sets.

2.1 Document Processing

2.1.1 Segmentation

Document text is segmented into sentence-level segments using the parser component of the spaCy[2] English large language model[3]. Sentence segmentation boundaries are the default punctuation characters defined by spaCy with the addition of semi-colons.

A document is therefore a container for a set of segments. Segment identifiers comprise the ID of the segment's document and an integer value indicating the segment's ordinal position in the document. The complete text of a document can be recreated by combining the segments in order although formatted structure (headers, lists, etc.) is lost.

Segments inherit the metadata of their document container, for example, peace agreement date, stage, or region. Date is necessary to measure the diffusion of a topic across a time series of documents.

2.1.2 Encoding

Encoding vectors are obtained from the USE model for all qualifying text segment. To qualify for encoding, a segment's text must exceed a minimum word count.

2.2 Matrices

2.2.1 Similarity Matrices

A similarity matrix comprises text segments in rows and columns, and cells that contain the semantic similarity of a row–column segment pair.

Matrix rows map onto an indexed set of text segment identifiers:

$$R = \{r_1, r_2, \ldots, r_M\} \tag{1}$$

[2] https://spacy.io
[3] https://github.com/explosion/spacy-models/releases/tag/en_core_web_lg-3.7.1

and their encoding vectors

$$\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_M\} \tag{2}$$

where M is the total number of row segments. The row segments may come from one or more document sets.

Matrix columns map onto an indexed set of text segment identifiers:

$$C = \{c_1, c_2, \ldots, c_N\} \tag{3}$$

and their encoding vectors

$$\boldsymbol{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_N\} \tag{4}$$

where N is the total number of column segments. The column segments may come from one or more document sets.

The semantic similarity score $s(r_m, c_n)$ of two text segments (r_m, c_n) is measured as the inverse of the angular distance between the encoding vectors of the two segments.

$$s(r_m, c_n) = 1 - \frac{\arccos\left(\dfrac{\boldsymbol{r}_m \cdot \boldsymbol{c}_n}{||\boldsymbol{r}_m||\,||\boldsymbol{c}_n||}\right)}{\pi} \tag{5}$$

The similarity matrix $\boldsymbol{S}$

$$\boldsymbol{S} = \begin{bmatrix} s(r_1, c_1) & s(r_1, c_2) & \cdots & s(r_1, c_N) \\ s(r_2, c_1) & s(r_2, c_2) & \cdots & s(r_2, c_N) \\ \vdots & \vdots & \ddots & \vdots \\ s(r_M, c_1) & s(r_M, c_2) & \cdots & s(r_M, c_N) \end{bmatrix}$$

is generated by computing the semantic similarity of every pair of row and column segments.

A similarity matrix can represent any number (k) of document sets as a complete undirected k–partite graph where graph vertices are text segments and edges connect the segments from the different sets. There are no edges between vertices within a document set.

The selection of the shape (number of rows and columns) of a similarity matrix is based on operational factors related to matrix size and computation times, as well as the research questions being asked. For example, when looking for topics in a single document set, the similarity matrix is a square symmetric matrix representing a complete unipartite graph ($k = 1$) comprising a single set of vertices (segments) with edges connecting every segment pair. Edges are weighted by the semantic similarity score $s_{m,n}$ of a pair.

When looking for topics across two sets of documents, we can put the segments of one document set in rows and the segments of the other document set in columns. The similarity matrix represents a complete bipartite graph ($k = 2$) comprised of two sets of vertices. If we also wanted topics within each document set then it would be necessary to construct a square symmetric matrix by concatenating the segments of the two document sets into a single set of segments which form both the rows and columns of the similarity matrix.

The complexity of matrix design increases when dealing with more than two document sets, i.e., $k > 2$. If only topics across document sets are required, then building separate similarity matrices for pairwise combinations of document sets is the most efficient method. The separate similarity matrices can be combined into an *adjacency matrix* (see below) later in the processing pipeline.

2.2.2 Thresholded Similarity Matrices

A similarity matrix is a dense matrix with a non-zero semantic similarity score in every cell. In many cases, pairs of segments are not semantically similar and therefore, the vast majority of cell values are low – see Figure 1 below. To remove these low value pairings, and prepare the matrix for downstream graph-based processing, the similarity matrix is thresholded.

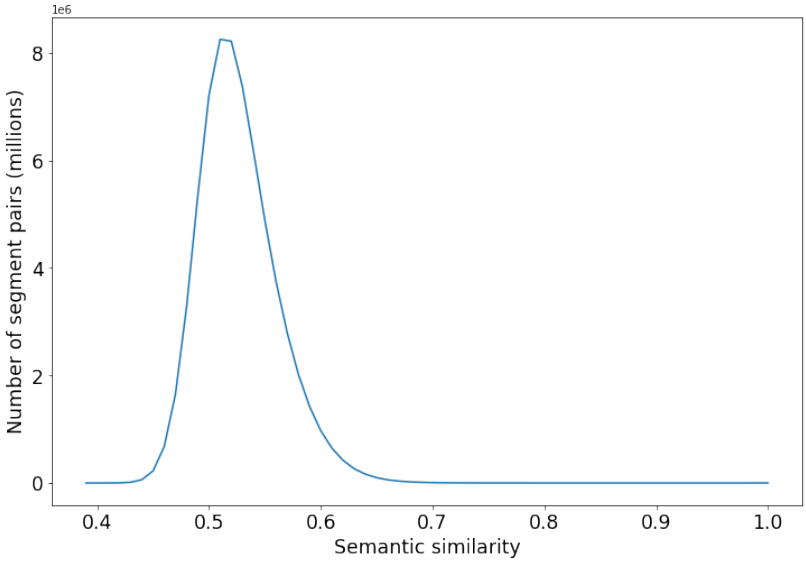

Figure 1: The distribution of the number of segment pairs (in millions) over semantic similarity scores. The similarity scores are from a similarity matrix built with 11493 encoded segments from PA-X v7 ceasefire agreements in both rows and columns.

A threshold is applied to convert a similarity matrix $\boldsymbol{S}$ into a binary-valued matrix $\boldsymbol{U}$ as follows:

$$u_{m,n} = \begin{cases} 0 & if\ \theta_{min} > s_{m,n} > \theta_{max} \\ 1 & if\ \theta_{min} \leq s_{m,n} \leq \theta_{max} \end{cases} \tag{6}$$

where θ_{min} is a minimum similarity threshold and θ_{max} a maximum similarity threshold such that $0 > \theta_{min} <= \theta_{max}$ and $\theta_{min} <= \theta_{max} <= 1.0$. The special case of $\theta_{min} = \theta_{max} = 1.0$ can be used to find topics comprising identical text segments.

A thresholded matrix represents an undirected graph where the value 1 indicates a connection between a row segment and a column segment[6]. Low values of θ_{min} will generate dense graphs containing many connections between semantically unrelated segments. Higher values of θ_{min} will generate sparser graphs with fewer connections between segments – see Figure 2 below.

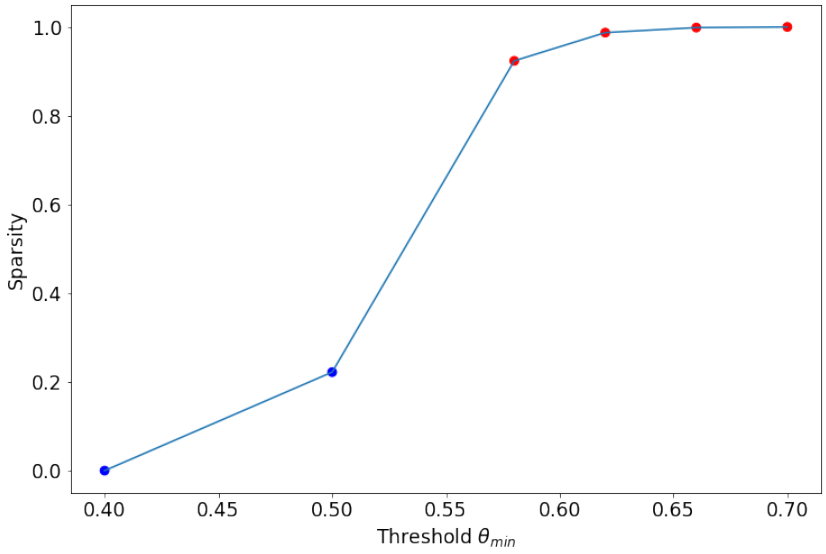

Figure 2: The sparsity of a thresholded similarity matrix at several values of θ_{min} with $\theta_{max} = 1$. The red dots are the values of θ_{min} used in Figure 3 below. The similarity matrix built with 11493 encoded segments from PA-X v7 ceasefire agreements in both rows and columns – see Figure 1 above.

2.3 Topic Discovery

Topic discovery is a two stage process:

1. Conversion of a thresholded similarity matrix into an undirected graph.
2. Finding the connected components of the graph using the Python SciPy API[4]

A connected component of an undirected graph is a disjoint set of vertices. Vertices that do not belong to a connected component are referred to as singletons. For a thresholded similarity matrix, a connected component identifies a cluster of semantically similar text segments.

Connected component analysis of a graph requires a square adjacency matrix. If row and column segments sets are identical, $R = C$, the thresholded matrix is already an adjacency matrix. If $R \neq C$ the matrix is a *biadjacency matrix* representing a bipartite graph.

[4] scipy.sparse.csgraph.connected_components

In order to find connected components in a biadjacency graph, the graph's biadjacency matrix must first be converted to an adjacency matrix. The relationship of a biadjacency matrix $\boldsymbol{B}$ and a corresponding adjacency matrix $\boldsymbol{A}$ is shown below.

$$\boldsymbol{A} = \begin{bmatrix} 0_{|Y|,|Y|} & \boldsymbol{B} \\ \boldsymbol{B^T} & 0_{|X|,|X|} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

The biadjacency matrix is contained in the upper quadrant of the adjacency matrix. The row and column vertices of the adjacency matrix are the concatenated vertices of the Y (row) set and X (column) set of $\boldsymbol{B}$: $\{y_1, y_2, y_3, x_1, x_2\}$

Figure 3 below illustrates the effect of θ_{min} on the connected components of a bipartite graph generated from a small similarity matrix comprising the segments on one document in rows and the segments of another document in columns. At the lowest value $\theta_{min} = 0.58$ the segments form a single component. As θ_{min} increases, connected components form and many segments become singletons which are not shown.

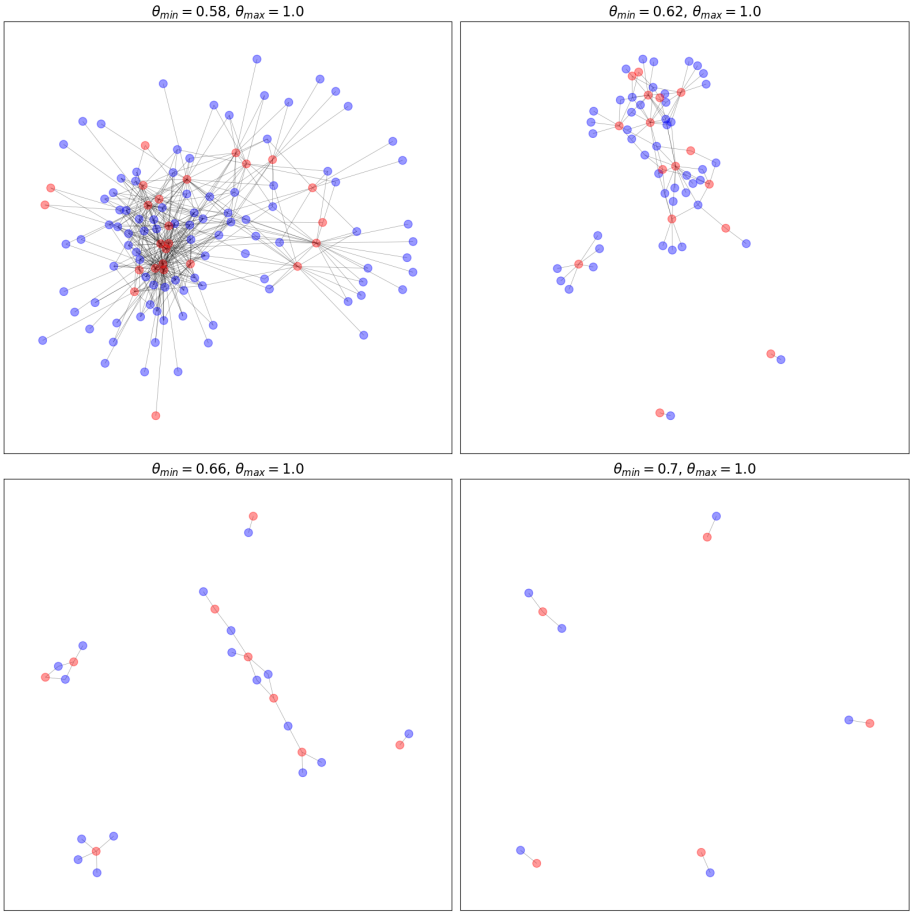

Figure 3: Bipartite graphs linking the segments of two documents for different values of θ_{min}. The blue vertices represent the segments in rows, and the red vertices the segments in columns. Singletons, i.e., segments unconnected to other segments are omitted.

Appendix A below provides links to example topics, and describes the topic acceptance and validation process.

2.3.1 Topic Summarisation

Summarisation of a topic uses a simple degree centrality algorithm to find the segment or segments in a topic with the most connections to other segments in the topic. If a topic spans two or more document sets, then the topic summary contains a segment from each set – see Appendix A below for sample topics and summaries.

Summary segments are used in downstream processing, for example, as templates for topic search, and for designing new topics for use in automatic classification of document text. They can also provide the basis for accepting or rejecting – see Appendix A – a candidate topic where the candidate topic comprises a very large number of segments.

2.3.2 Diffusion of Topics

Figure 4 is a visualisation of the diffusion over time of the 162 topics that the *Revitalised Agreement on the Resolution of the Conflict in the Republic of South Sudan (R-ARCSS)* has in common with the 68 implementation reports that follow it. Implementation reports include those from JMEC, RJMEC, and CTSAMVM. A dot represents a report containing a topic. The dense vertical line at the left of the diagram is the *JMEC-1st-Qtr-2020-Report-FINAL 1* report which contains 81 of the possible 162 topics.

The diagram provides information about:

- Which topics are mentioned most.
- Which reports mention the most topics.
- When a topic first appears in the report sequence.

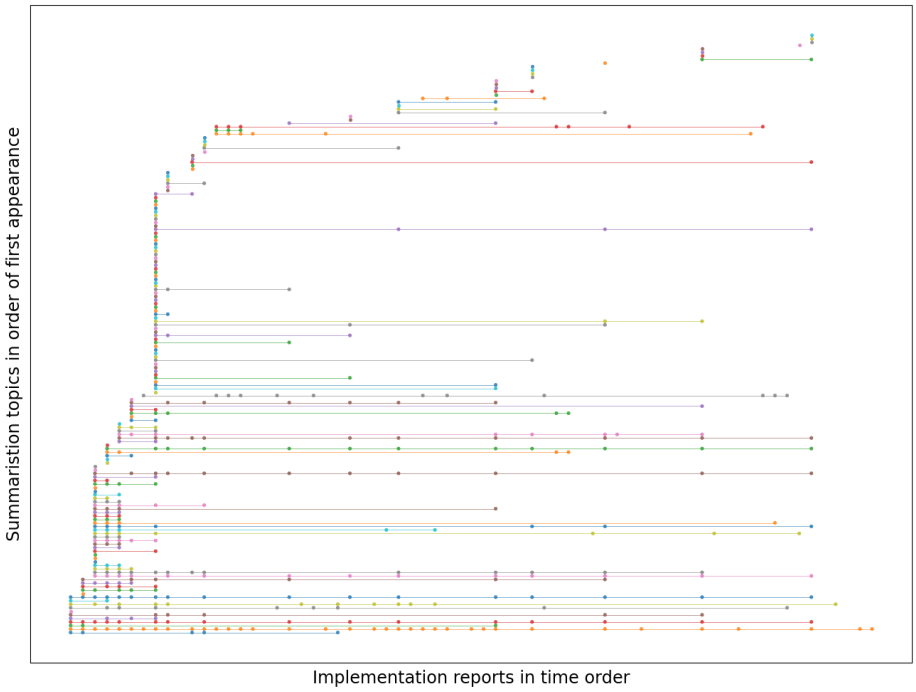

Figure 4: Diffusion diagram showing topics in the Revitalised Agreement on the Resolution of the Conflict in the Republic of South Sudan (R-ARCSS) has in common with the 68 implementation reports that follow it. A horizontal line represents a topic, and a dot is present when the topic appears in a report. Reports are organised in time order along the x-axis.

3 Discussion

3.1 Relationship to Topic Modelling

Methods based on matrix factorisation, for example Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorisation (NMF), are popular topic modelling techniques[5][1]. These methods seek to find latent features (topics) in a set of documents represented by a word–document matrix[5].

[5]When multiplying matrices, the dimensions of the factor matrices may be significantly lower than those of the product matrix. It follows that factorising a matrix can produce factors with significantly reduced dimensions compared to the original matrix. For example, if $\boldsymbol{V}$ is an $m \times n$ matrix and we are looking for p topics, the factor matrix $\boldsymbol{W}$ (known as the features matrix) is an $m \times p$ matrix, and the factor matrix $\boldsymbol{H}$ (known as the coefficients matrix) is a $p \times n$ matrix. The dimension p corresponds to the topics. The assumption is that that each column in the product matrix $\boldsymbol{WH} \approx \boldsymbol{V}$ is a linear combination of the p column vectors in the features matrix $\boldsymbol{W}$ with coefficients supplied by the coefficients matrix $\boldsymbol{H}$.

End-users are presented with lists of weighted words and their task is to interpret these lists as topics. Based on the author's experience of using these techniques in commercial settings, issues are:

- Documents and topics are represented as lists of context-free words.
- As the number of requested topics increases, the topic word lists overlap.
- The technique performs poorly with homogenous document sets.
- Topic model users have to make a series of decisions, for example, pre-processing the text, selecting the number of topics, and interpreting the topics that can dramatically alter the final results[7][4].

At best, the technique may provide broad-brush categorisation of disparate sets of documents.

In contrast, the method described in this report represents topics as collections of semantically similar sentences or clauses that can be tracked across document time series. In the current implementation, the user's only pre-processing decision is the choice of the θ_{min} and θ_{max} thresholds. Once topics have been generated, the acceptance or rejection of a topic is based on the reading of real sentences supported by document metadata – see Appendix 1.

3.2 Limitations

The current implementation requires a user to select θ_{min} and θ_{max} thresholds. In some situations, several runs at different values θ_{min} may be required to capture all topics, and topics may disappear or fragment depending on the value of θ_{min}. In such a situation, the user must combine results from the outputs generated at the various values of θ_{min} before moving to the acceptance/rejection stage. We are currently exploring automation of the generation and combination of topics from a range of θ_{min} values.

A Topics Discovered in UNMISS Reports

A.1 Topic Discovery Results

The file topics.xlsx contains results obtained from a randomly selected set of 10 UNMISS reports[6]. Threshold values were $\theta_{min} = 0.7$ and $\theta_{max} = 0.98$.

[6] https://unmiss.unmissions.org/human-rights-reports

Altogether, 82 candidate topics were found, 20 of which were rejected, leaving 62 substantive topics. Candidate topics containing section headings, dates, etc. were judged to be irrelevant and were rejected. Colours are used to identify accepted topics. Rejected topics are listed at the end of the file.

The following conditions apply:

- The segments of a topic are in ascending order of document date which provides data on the diffusion of the topic across time.
- A topic's numerical ID is in the first column.
- A summary segment is marked by the value TRUE in the second column.
- A human makes the decision whether to accept a topic – see the *Accept topic* column.

A.2 Validating Topics

To further check the validity of an accepted topic, the topic's summary segment is used to search for similar segments in the set of documents.

For example, the file incidents.xlsx contains the search results for the summary segment from topic ID 11 in the topics.xlsx file: *All incidents, particularly those involving intercommunal violence, are deconflicted with incidents documented by the UNMISS Civil Affairs Division (CAD)*

All five segments of the topic are found (in yellow) along with an additional 13 similar segments. Importantly, relevant segments were found relating to 'sub-national violence' and 'community-based militia'.

The validation stage is important because topics are generally obtained using high semantic similarity thresholds and, as demonstrated in incidents.xlsx, this may mean that less similar, but nonetheless significant segments, are not present in a topic's segment set.

B References

[1] Charu C. Aggarwal. *Machine Learning for Text*. Springer, Cham, second edition, 2022.

[2] Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder, 2018.

[3] Cruz A, Elkins Z, Gardner R, Martin M, Moran A. A new method for analyzing public consultation data. *PLoS ONE*, 18(12): e0295396, 2023.

[4] Matthew J. Denny and Arthur Spirling. Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political Analysis*, 26(2):168–189, 2018.

[5] Thiago Faleiros and Alneu Lopes. On the equivalence between algorithms for non-negative matrix factorization and latent dirichlet allocation. In *ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.*, 2016.

[6] Mark Newman. *Networks*. Oxford University Press, 2018.

[7] Simmons, J. P., Nelson, L. D., & Simonsohn, U. False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11):1359–1366, 2011.