Discovering the Geometry of Narratives and their Embedded Storylines

Eduard Hoenkamp[1,2]
[1]*Science and Engineering Faculty, Queensland University of Technology (QUT), Brisbane, Australia*
[2]*Institute for Computing and Information Sciences, Radboud University, Nijmegen, The Netherlands*

Keywords: Storyline, Topic Models, Document Space, Foreground/Background Separation, Robust PCA, Sparse Recovery, Subspace Tracking, Geometric Optimization, Grassman Manifolds.

Abstract: Many of us struggle to keep up with fast evolving news stories, viral tweets, or e-mails demanding our attention. Previous studies have tried to contain such overload by reducing the amount of information reaching us, make it easier to cope with the information that does reach us, or help us decide what to do with the information once delivered. Instead, the approach presented here is to mitigate the overload by uncovering and presenting only the information that is worth looking at. We posit that the latter is encapsulated as an underlying *storyline* that obeys several intuitive cognitive constraints. The paper assesses the efficacy of the two main paradigms of Information Retrieval, the document space model and language modeling, in how well each captures the intuitive idea of a storyline, seen as *a stream of topics*. The paper formally defines *topics* as high-dimensional but sparse elements of a (Grassmann) manifold, and *storyline* as a trajectory connecting these elements. We show how geometric optimization can isolate the storyline from a stationary low dimensional story background. The approach is effective and efficient in producing a compact representation of the information stream, to be subsequently conveyed to the end-user.

1 INTRODUCTION

In today's world, news feeds may become obsolete in minutes, e-mails stack up, and fresh tweets may arrive before we have digested the current one. Just back from a vacation, or after having been away from the internet for some time, we have to rely on our friendly neighbor or colleague for a summary of events, in order to resume absorbing those information feeds. Absent such human helper, good algorithms for summarization and topic tracking are called for to keep abreast of all those events.

In this paper we like to investigate which, if any, traditional IR techniques can be used for this task. But first we want to distinguish techniques whose value lie in detecting and tracking topics in real-time, hence under time pressure, from the more fundamental question of what exactly it is that we would like to track. To this end we will study a case far away from the maddening internet. The case where people have control over the pace at which they process the stream of topics they encounter: reading a book at leisure.

As illustration we will use Hemingway's *The Old Man And The Sea* for which he was awarded a Nobel prize. The story is short, simple, and likely familiar to many readers. Note that we won't give a formal definition of 'storyline' because we have none, and defining it as a 'plot' would only beg the question. But the reader will probably agree that for *The Old Man and the Sea* it will be somewhere between "a man catches a big fish" which has too little detail, and the book itself which has too much.

Furthermore, while reading the literature, we did not come across any a priori constraints on the concept of a storyline. So let us mention some *cognitive constraints on the model* that seem so self-evident that we might otherwise forget to incorporate them in the model[1]. While reading a book, we can see ourselves:

CC1: Skim a page without losing the storyline
CC2: Recount the storyline after one read
CC3: Ignore frequent words without loosing track
CC4: Recount the storyline thus far
CC5: Encounter generally more words than topics

[1]We invite the reader who disagrees with some of these items to check at the end of the paper if they would have consequences for the model. Elsewhere we already demonstrated the use of other cognitive constraints on improving existing models, e.g. in the area of information overload (Hoenkamp, 2012) and language modeling (Hoenkamp and Bruza, 2015)

Hoenkamp, E.
Discovering the Geometry of Narratives and their Embedded Storylines.
DOI: 10.5220/0008356004830490
In *Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2019)*, pages 483-490
ISBN: 978-989-758-382-7

Table 1: LDA topic representation for *The Old Man and the Sea* where the narrative is treated as corpus with the pages as documents. (a) the top ten topics, with words in order of probability (b) pages best described by the topics on the left (c) the first topic as a 'word cloud', where the size of a word is relative to its probability in the distribution.

Topics	Pages described by this topic	A 'word cloud' for the first topic
shark hit bring club close pain drove	28, 30, 32, 31, 33, 10, 8	
took course know inside set beat live	20, 4, 28, 29, 6, 21, 12	
ask mast basebal father high today eighty	5, 3, 19, 2, 29, 4, 10	
think aloud knife meat seen blade sorry	31, 29, 30, 32, 5, 15, 28	
dark fish wood purple cut soon cord	20, 22, 7, 18, 15, 17, 9	
head let felt turn side put feel	26, 24, 11, 30, 25, 32, 33	
watch light fly night stern dolphin left	7, 8, 20, 21, 33, 10, 22	
eat left hour cramp steady open arm	16, 11, 12, 15, 17, 22, 21	
boy remember carry strong road bed tell	6, 3, 4, 35, 1, 34, 2	
water thought fast circle bow yellow rope	9, 28, 8, 25, 17, 27, 26	
(a)	(b)	(c)

To make headway we introduce our working definition of a *storyline as the sequence of topics* in the narrative. And in the terminology of traditional Information Retrieval (IR), we use the book as corpus, and the pages in the book as documents. So doing, they are amenable to the same techniques when needed[2] in both of the main paradigms of IR. We will first look at probabilistic language models, and then turn our attention to the document space approach. (One motivation to elaborate both was to avoid someone asking why we did not study the other other paradigm.)

2 LANGUAGE MODELING

With the substitution of pages and book for documents and corpus, we will first follow the IR model of *Latent Dirichlet Allocation (LDA)* but applied to pages in a book. In this model, a *topic* is defined as a probability distribution over words. Recall that LDA postulates the probability distribution over topics θ, topic vectors $\mathbf{z}$, and word vectors $\mathbf{w}$ as follows:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha)\prod_{i=1}^{N} p(z_i|\theta)p(w_i|z_i, \beta) \quad (1)$$

where the probability right after the equal sign is a Dirichlet distribution and the next two are multinomials. We ran the LDA model on *The Old Man and the Sea*, resulting in the summary of Table 1. At first sight this looks good. The first topic has, among others, to do with 'shark', and in the the middle column we see what pages it applies to. This can be checked with the book in hand. The topic is indeed important in the storyline at the end of the book (pages 28 to 33 as found by LDA). It is also significant that some pages where the word 'shark' occurs are not mentioned (pages 2, 7, 9 and 14), and that these pages are indeed without import for the storyline. Unfortunately, what works for 'shark' is very difficult to replicate for other topics. Another problem is that the outcome seems to be the luck of the draw: when we changed the number of topics anticipated in LDA from 25 to 10, we got as first topic {*sea fast water turn eye bird circle bait*}, the 'shark' topic became less prominent and changed to {*shark head skiff kill aloud hit oar saw*}, and the other topic distributions became even more unintelligible as a storyline. (Note that by the same token, this makes it hard to check the cognitive constraints **CC1** and **CC3** we set out in the introduction. The others seem to hold.) But it was already well-known in the LDA literature that topics as a list of words are hard to interpret. This is akin to the situation with uninterpreted latent factors in LSA (C.Deerwester et al., 1989). And like the latter, it does not prevent LDA from being successfully applied in classifying documents correctly, assigning authors, or analyzing shift

[2]Instead of pages one can think of other units, such as chapters, sections, and paragraphs, as we will see.

in topic (Griffiths and Steyvers, 2004).

Not only did we vary the LDA parameters in our study, we also experimented with amending the definition for the LDA probability distribution in equation 1. After all, in the current form it does not incorporate the story's continuity over subsequent pages. The generative process starts anew for each document, selecting a topic distribution irrespective of the document chosen previously. Hence each page is also generated anew without regard of the previous page, i.e. ignoring the continuity of the story. However, amending the distribution did not help. It seemed to usher us back in the direction of *pLSA* (Hofmann, 1999) which LDA so successfully superseded. Then we experimented with proposals in the literature about on-line LDA, for example using 'empirical Bayes' techniques (AlSumait et al., 2008) or approaches to detect topic drift by identifying change in Z-score for central tendency (Wilson and Robinson, 2011). Perhaps these approaches require larger samples from independent distributions, which does not apply to book pages. In brief, our attempts to use Language Modeling to discover a storyline seems to have reached a dead end. But we are not alone in this conclusion. The NIST sponsored *topic detection and tracking (TDT)* initiative "has ended and will not be restarted in the near future" (Allan et al., 1998). Therefore, and for the time being, we decided to give up on the language modeling approach, report our results here[3], and move on to the other important IR paradigm, the vector space model for documents. Or rather, we will use a more abstract topological extension of it.

3 THE DOCUMENT SPACE

We will now continue with the standard representation of the document space, namely the term-by-document matrix. Recall that the rows are indexed by words and the columns by documents. The entries are numbers usually weighted according to term frequency and inverse document frequency. The columns can be viewed as vectors in a high-dimensional space with the words as basis vectors.

[3]Although it is usually difficult to get negative results published, it is important to try nonetheless. It is necessary to prevent other researchers from wasting time doing the research all over, and it is crucial to counterbalance positive results that others published and which might be statistically significant only in isolation.

3.1 Revisiting Luhn 1957 and 1958

In his groundbreaking work at the end of the 1950's, Luhn (Luhn, 1957) described a number of document preparation steps, such as term frequency normalization, stop-list removal, stemming, and the use of thesauri. These steps have persisted in IR over these six decades. Lesser known seems his work on the *Automatic Creation of Literature Abstracts* (Luhn, 1958), of which the objective was "to save a prospective reader time and effort in finding useful information" especially as the "widespread problem is being aggravated by the ever-increasing output" (p. 159). This is a similar objective as given in recent IR proposals for storyline extraction, namely to reduce information overload. Several of the recent proposals even contain some of Luhn's mechanisms. Oddly, as the reader can verify, references to this work are glaringly absent in those proposals.

In order to recover a storyline or produce a summary, two steps must be taken (1) detect topics and (2) output a representation for each topic. In so-called *extractive summarization*, the first step locates sentences in a document which are concatenated into a summary (see (Saggion and Poibeau, 2013) for an overview).

Luhn (Luhn, 1958) proposes a method he calls 'auto-abstract' which first computes a significance value for sentences based on word frequencies and word proximities. The significant sentences that rank highest are then output to form an abstract (i.e. extractive summarization). If one adds an extra step, namely notice when the vocabulary changes substantially over significant sentences, this can be used to locate topic boundaries. This is essentially the method

Table 2: A "movie after the book" (a) Depiction of a word-by-page matrix. (b) Each column (page vector) is folded into a frame with entries normalized to values of gray scale pixels. This metaphor helped to experiment with algorithms developed for surveillance videos as a way to divide text on a page into *background* and *foreground*; the latter to represent topics that occur on the page.

01	21	etc.	—
02	22	—	—
03	23	—	—
04	24	—	—
05	25	—	—
06	26	—	—
07	27	—	—
08	28	—	—
09	29	—	—
10	30	—	—
11	31	—	—
12	32	—	—

(a)

$\Longrightarrow$

01	02	03	04
05	06	07	08
09	10	11	12

21	22	23	24
25	26	27	28
29	30	31	32

etc. . . . . .
. . . . . .
. . . . . .

(b)

used for the much more recent technique of *TextTiling* (Hearst, 1997). Other algorithms do not just notice vocabulary changes, but changes in vocabulary distribution to delineate topic boundaries (Mao et al., 2007). The approach in this paper is in spirit akin to Luhn's. However topics are located based on the geometry of the document space, as we will see, where the documents will be the pages in the book.

3.2 A Foreground/Background Analogy

The techniques in the remainder of this paper are best introduced by way of their analogy to video processing for surveillance cameras. The task there is to separate foreground from background, e.g., to discover an intruder against the background of a lobby. Now imagine we equate storyline with foreground, and the uninteresting part of a page with background. Just as videoprocessing lets the intruder stand out from the lobby, we can explore similar techniques to let the storyline stand out from the rest of the story. (Readers should recall this metaphor when they would get lost in technicalities later in this paper.) Given the success of such algorithms for video surveillance, we adapted a number of such algorithms to bring the storyline to the fore, as we will explain in a moment. To explore if there were algorithms suitable for our purpose (i.e. applied in the linguistic domain) we transformed a book into a video as follows: Start with a word-by-page matrix and normalize the entries to grey-scale pixels. Next, factor the number of words in two numbers, say l and w and make a rectangle of hight l by width w. Now fill the rectangle from top to bottom with the first column of the word-by-page matrix and repeat this for all columns, see table 2. This way, we can view the sequence of rectangles as the frames of a movie and, presto, all well-known algorithms developed for video are available to process the word-by-page representation as a sequence of video frames. So next we will explain the foundations of a class of successful video algorithms for foreground/background separation and show that these indeed fare well in the analogical case for storyline discovery[4].

[4]Of course analogies and metaphors are often helpful in research. Many years ago, at the time that Latent Semantic Analysis (LSA) was studied intensively, the main technique for dimension reduction was Singular Value Decomposition (SVD). At that time I proposed to represent the word-by-document matrix as a picture, hence making it amenable to a plethora of image processing techniques. For LSA this resulted in many efficient alternatives for SVD, most notably JPEG 2000 (Hoenkamp, 2003).

4 GEOMETRIC OPTIMIZATION

The remainder of this paper assumes familiarity with *dimension reduction*, and the reader familiar with that concept can skip ahead to section 4.2. First we will briefly take a step back, from processing the high dimensional document space to the mundane example of linear regression. Suppose we have a collection of data that we plot as points in a 2-D graph. Informally, linear regression is a way to draw a straight line through the data points such that it best fits the data. Formally it is a way to reduce a two dimensional space (the points in the 2-D graph) to a one-dimensional space (the straight line) that is nearest to it in terms of Euclidian distance. (Hence also the name 'least-squares' method.) In higher dimensions, such dimension reduction is usually achieved through *Principle Component Analysis* or PCA, which is also a least-squares method. Given a measurement matrix M the data model is $M = L_0 + N_0$ where L_0 is low rank and N_0 a small perturbation matrix representing noise. PCA estimates L_0 by a k-dimensional approximation L in the sense of least-squares, i.e. the Euclidean norm $||.||_2$ is minimized as follows:

$$\begin{array}{ll} \text{minimize} & ||M-L||_2 \\ \text{subject to} & rank(L) \leqslant k \end{array}$$

The approach is known to be very sensitive to *outliers*. Outliers therefore usually receive special treatment in data analysis, sometimes by explaining them away, or by removing them from consideration.

Whatever the treatment, outliers are usually to be avoided. But this requires a method to locate the pesky outliers in the first place. In a graphical representation one can rely on visual inspection, but in higher dimensions this is not so straightforward. Not long ago, an effective approach to the problem of locating outliers has been proposed in the form of *Robust PCA*, which has been developed in the area of *Compressed Sensing* (CS). To our knowledge, and consulting recent literature, ours is the first time that CS is applied to language processing. Compare the comprehensive overview of (Bouwmans et al., 2018) where this approach is not to be found among the large number of application areas.

4.1 The Storyline as a Sparse Subspace

Continuing the surveillance video metaphor, unless something eventful happens, such as a burglar entering the premises, each frame consists of thousands of pixels highly correlated with the next frame. Consequently, these data form a very low dimensional subspace of the high dimensional space of all possible

pixel combinations. Similarly, imagine it were possible to leave out the storyline in a narrative, not much would remain other than a boring list of words, highly correlated from one page to the next. In other words, *in the case of the storyline the outliers are the objects of interest,* representing the topics. So instead of trying to remove the outliers from the data as noise, we want to keep them in.

Formally, we want to split the word-by-page matrix M from the narrative as the sum of a low dimensional matrix L_0, and a high dimensional but sparse matrix S_0 of topics (the spikes as it were in the otherwise boring story):

$$M = L_0 + S_0$$

In signal analysis, as with many statistical problems, one is interested in finding L_0 from the measurements M, where S_0 forms the noise that one wants to get rid off.

So the focus is on isolating and possibly removing such outliers. But again, in our case the outliers are the objects of interest. In other words, we want to recover S_0 from the data M. This, however, is a severely under-constrained problem, as there is a potentially infinite number of ways to split the matrix M such that $M = L + S$. So how can this ever be accomplished? For this we will turn to a curious result in the blossoming field of *compressed sensing*, see e.g. (Baraniuk, 2007) for an introduction to the field.

4.2 Robust PCA

Again, what we are trying to do, is solving the seemingly impossible problem of recovering S_0 from the under-constrained equation $M = L_0 + S_0$. But a truly remarkable theorem was proven by Candès and colleagues (Candès et al., 2011) namely that under some (precisely defined) assumptions, it is indeed possible to recover both the low-rank and the sparse components exactly. The algorithm they propose is a convex program called *Principal Component Pursuit* which solves the problem (Candès et al., 2011):

$$\begin{array}{ll} \text{minimize} & ||L||_* + \lambda||S||_1 \\ \text{subject to} & M = L + S \end{array}$$

with trace norm $||.||_*$ and l_1 norm $||.||_1$. How do we know that there is a solution to this optimization problem in the case of storyline discovery?
Recall that the computation requires two steps:

1. Dimension reduction, and
2. Locating the outliers

Regarding the first item: There are many ways to accomplish this, traditionally through *PCA* and the related *SVD* (singular value decomposition), founded on basis transformations. In our work we use the more recent technique of *random projections* (Vu et al., 2018; Bingham and Mannila, 2001). A problem could arise if dimension reduction resulted in basis transformations that destroy the constellation of the storyline in the manifold. For example, it could change the order of events in the story, and that is not what we want. Fortunately we can rely on the following lemma (Johnson and Lindenstrauss, 1984):

Lemma. *For* $0 < \varepsilon < 1$*, any n, and* $k \geq \frac{24}{3\varepsilon^2 - 2\varepsilon^3} \log n$ *then for any set A of n points* $\in \mathbb{R}^d$ *there exists a map* $f : \mathbb{R}^d \to \mathbb{R}^k$ *such that for all* $x_i, x_j \in A$
$(1-\varepsilon)||x_i - x_j||^2 \leq ||f(x_i) - f(x_j)||^2 \leq (1+\varepsilon)||x_i - x_j||^2$

This, in other words, guarantees that there exists a linear operator that leaves the distances between pairs of points approximately in tact. Since the lemma is independent of the dimension of the original space, in the present application it does not depend on the size of the lexicon. But knowing there exists a solution is different from finding one.

Regarding the second item: Once the dimension reduction has been performed, the outliers are to be found in the space orthogonal to the low dimensional space. That is, once the storyline has been separated from the background noise, the remaining part of the manifold contains the uninteresting part, the glue between the sequence of interesting events.

If we represent the book in its entirety as a manifold of dimension n, then the algorithms reconstruct a sequence of sub-manifolds S of dimension say m, forming the Grassmann manifold $Gras(m,n)$[5], which the physicist reader may recognize from String theory (Schwarz, 1999). Given that the sequence of solutions S represents the storyline, one could express discovering a storyline as tracking the m-dimensional topics in an n dimensional Grassmann manifold representing the book[6].

After so much theory it is time to see how this works out in practice.

4.3 Checking Cognitive Constraints

In section 2 we found it hard to see how the Language Modeling paradigm could comply with the cognitive

[5] $Gras(m,n)$ is the collection of manifolds of dimension m contained in a manifold of dimension n, which is not necessarily a Hilbert space as is IR's traditional document space.

[6] For the reader who could use a more concrete mental picture of this approach, we refer to an application in the area of emotion detection (Alashkar et al., 2018) mainly because it contains illustrations that may help envisage the technique.

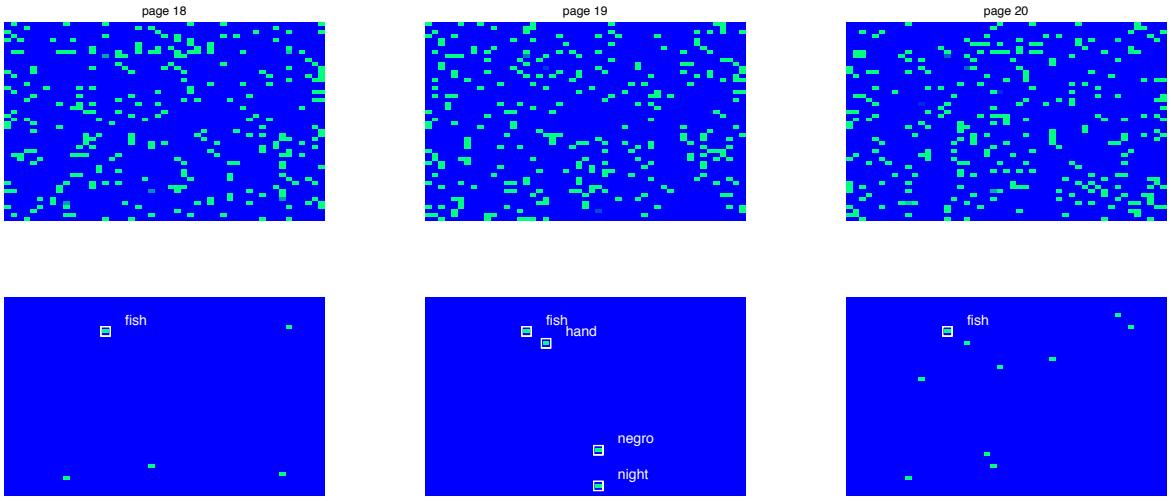

Figure 1: Topics appearing and disappearing in subsequent pages halfway *The Old Man and the Sea* (pages 18, 19, 20). Top: Page vectors folded into 'video frames.' Bottom: the same folded pages after Robust PCA. Some topics persist over several pages, such as 'fish' in the upper left. Others are short-lived, such as the topic {*hand, negro, night*} of page 19, where the old man recounts how he "had played the hand game with the great negro from Cienfuegos who was the strongest man on the docks. They had gone one day and one night with their elbows on a chalk line on the table and their forearms straight up and their hands gripped tight. Each one was trying to force the other's hand down onto the table".

constraints set out in the introduction. Hence we need to check if the Document Space paradigm fares any better in this respect.

Applications of compressed sensing resulted in a variety of algorithms to isolate sparse subspaces. Another result is that a matrix of type L, i.e. low rank and dense, can be reconstructed even if data M is highly corrupted or when there are many missing data. This is precisely the case for cognitive constraint **CC1**. So the good news is that even if many words are ignored, that is, when there are missing data in M, Principal Component Pursuit can still reconstruct both L_0 and S_0. So when **CC2** is satisfied, say when a speed reader can recall a storyline, the algorithm can reconstruct the storyline as well. This also applies to **CC3**, which means that the narrative can be processed as is custom in IR and the topics can still be discovered. Finally, for algorithms as RPCA it is known what degree of sparsity is needed for it to find a solution to the objective function to be optimized. The degree of sparsity in the language domain depends on the proportion of topics to words. And that the number of topics is usually much smaller than the number of words used to convey these topics is an instance of **CC5**. That constraint guarantees that there must exist a sparse subspace for the storyline and the Johnson Lindenstrauss lemma even defines the degree of sparsity attainable.

4.4 Results

Our transformation of a story to a video sequence allowed us to experiment with a plethora of algorithms for foreground/background separation. So new algorithms from the video processing field using geometric optimization, can be incorporated as well. These fall largely in two categories: some researchers optimize for $M = L + S$ (Hage and Kleinsteuber, 2013; Seidel et al., 2014), others optimize for $M = L + S + D$ with error term D (Zhou and Tao, 2011). The technical details of these algorithms are beyond the scope of this paper. For a comprehensive overview of these techniques we refer to (Bouwmans et al., 2018) and for Newton methods to solve the equations to (Edelman et al., 1998). But we do not want to leave the more application oriented reader empty-handed. Therefore we include as example the application of 'bilateral random projections' proposed in (Zhou and Tao, 2011) to *The Old Man and the Sea*. The result is depicted in Figure 1 for pages halfway the book[7]. We used several other books for our evaluation, namely *The Da Vinci Code* and the first volume of *The Lord of the Rings*. To compare the various methods, we obtained code (mostly Matlab) from the authors (see reference list), who were extremely helpful. Of course we needed to rewrite code that was written for the

[7]Note well that the frames are not word by document vectors, but each frame represents one document vector, rolled up as in Table 2

video domain and adapt it to the language domain. The resulting sparse S matrices were then (1) projected back into the word space (compare figure 1), (2) verbalized using extraction summarization (i.e. with their surrounding sentences) and placed one after another to form the storyline, (3) we asked colleagues to evaluate the storylines (e.g. (Janaszkiewicz et al., 2018)). In this informal evaluation the method of (Zhou and Tao, 2011) gave the best results.

5 CONCLUSION

To stem the information deluge, many researchers have proposed algorithms and techniques to mitigate the often overwhelming stream of information. These approaches are most often tailored to specific users, kinds of information, or circumstances, see the very comprehensive overview of (Strother et al., 2012). We take the view that different kinds of information streams, from news feeds, to mail exchanges, to twitterstorms, all keep the reader in suspense of the developing storyline. This allows us the unifying approach of studying how to capture such storylines. We presented the analogy of book pages to video frames, hence borrowed heavily from techniques from the processing of surveillance videos. We used the mathematics developed in the area of compressed sensing and showed how it can be applied in the linguistic domain for the discovery of storylines. We have not extensively experimented to validate the approach, but we showed that the sound underlying mathematics, the cognitive plausibility, and the informal experiments are promising and warrant further investigation.

REFERENCES

Alashkar, T., Amor, B. B., Daoudi, M., and Berretti, S. (2018). Spontaneous expression detection from 3D dynamic sequences by analyzing trajectories on grassmann manifolds. *IEEE Trans. Affective Computing*, 9(2):271–284.

Allan, J., Carbonell, J., and Doddington, G. (1998). Topic detection and tracking pilot study final report. In *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, pages 194–218.

AlSumait, L., Barbará, D., and Domeniconi, C. (2008). On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proc. 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 3–12, Washington, DC, USA. IEEE Computer Society.

Baraniuk, R. G. (2007). Compressive Sensing. *IEEE Signal Processing Magazine*, 24(118-120,124).

Bingham, E. and Mannila, H. (2001). Random projection in dimensionality reduction: Applications to image and text data. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 245–250, New York, NY, USA. ACM.

Bouwmans, T., Javed, S., Zhang, H., Lin, Z., and Otazo, R. (2018). On the applications of robust pca in image and video processing. *Proceedings of the IEEE*, 106(8):1427–1457.

Candès, E. J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *J. ACM*, 58(3):11:1–11:37.

C.Deerwester, S., Dumais, S. T., W.Furnas, G., Harshman, R. A., Landauer, T. K., Lochbaum, K. E., and Streeter, L. A. (1989). U.S. Patent No. 4,839,853. Washington, DC: U.S. Patent and Trademark Office.

Edelman, A., Arias, T. A., and Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *Siam J. Matrix Anal. Appl*, 20(2):303–353.

Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proc. National Academy of Sciences*, 101(5):5228–523.

Hage, C. and Kleinsteuber, M. (2013). Robust PCA and subspace tracking from incomplete observations using l_0-surrogates. *Computational Statistics*, 29(3):467–487.

Hearst, M. A. (1997). Texttiling: Segmenting text into multi-paragraph subtopic passages. *Comput. Linguist.*, 23(1):33–64.

Hoenkamp, E. (2003). Unitary operators on the document space. *Journal of the American Society for Information Science and Technology*, 54(4):314–320.

Hoenkamp, E. (2012). Taming the terabytes: a human-centered approach to surviving the information-deluge. In Strother, J., Ulijn, J., and Fazal, Z., editors, *Information Overload : A Challenge to Professional Engineers and Technical Communicators*, IEEE PCS professional engineering communication series, pages 147–170. John Wiley & Sons, Ltd, Hoboken, New Jersey.

Hoenkamp, E. and Bruza, P. (2015). How everyday language can and will boost effective information retrieval. *Journal of the Association for Information Science and Technology*, 66(8):1546–1558.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proc. 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50–57, New York, NY, USA. ACM.

Janaszkiewicz, P., Krysińska, J., Prys, M., Kieruzel, M., Lipczyński, T., and Różewski, P. (2018). *Text Summarization For Storytelling: Formal Document Case*, volume 126, pages 1154 – 1161. Elsevier.

Johnson, W. B. and Lindenstrauss, J. (1984). Extensions of lipschitz mappings into a hilbert space. In *Conference in modern analysis and probability*, volume 26, pages 189–206. Amer. Math. Soc.

Luhn, H. P. (1957). A statistical approach to mechanized

encoding and searching of literary information. *IBM J. Res. Dev.*, 1(4):309–317.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2(2):159–165.

Mao, Y., Dillon, J., and Lebanon, G. (2007). Sequential document visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1208–1215.

Saggion, H. and Poibeau, T. (2013). Automatic text summarization: Past, present and future. In Poibeau, T., Saggion, H., Piskorski, J., and Yangarber, R., editors, *Multi-source, Multilingual Information Extraction and Summarization*, pages 3–21, Berlin, Heidelberg. Springer.

Schwarz, A. (1999). Grassmannian and string theory. *Communications in Mathematical Physics*, 199(1):1–24.

Seidel, F., Hage, C., and Kleinsteuber, M. (2014). prost: A smoothed l_p-norm robust online subspace tracking method for background subtraction in video. *Mach. Vision Appl.*, 25(5):1227–1240.

Strother, J. B., Ulijn, J. M., and Fazal, Z. (2012). *Information Overload: An International Challenge for Professional Engineers and Technical Communicators*. Wiley-IEEE Press, 1st edition.

Vu, K., Poirion, P., and L., L. (2018). Random projections for linear programming. *Mathematics of Operations Research*, 43(4):1051–1071.

Wilson, A. T. and Robinson, D. G. (2011). Tracking topic birth and death in LDA. Technical report, Sandia National Laboratories.

Zhou, T. and Tao, D. (2011). Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In Getoor, L. and Scheffer, T., editors, *Proc. 28th Int. Conf. on Machine Learning (ICML-11)*, ICML '11, pages 33–40, New York, NY, USA. ACM.