Multivariate Powered Dirichlet-Hawkes Process

Gaël Poux-Médard[1][0000−0002−0103−8778], Julien Velcin[1][0000−0002−2262−045X], and Sabine Loudcher[1][0000−0002−0494−0169]

[1] Université de Lyon, Lyon 2, ERIC UR 3083, 5 avenue Pierre Mendès France, F69676 Bron Cedex, France
gael.poux-medard@univ-lyon2.fr
julien.velcin@univ-lyon2.fr
sabine.loudcher@univ-lyon2.fr

Abstract. The publication time of a document carries a relevant information about its semantic content. The Dirichlet-Hawkes process has been proposed to jointly model textual information and publication dynamics. This approach has been used with success in several recent works, and extended to tackle specific challenging problems –typically for short texts or entangled publication dynamics. However, the prior in its current form does not allow for complex publication dynamics. In particular, inferred topics are independent from each other –a publication about finance is assumed to have no influence on publications about politics, for instance.
In this work, we develop the Multivariate Powered Dirichlet-Hawkes Process (MPDHP), that alleviates this assumption. Publications about various topics can now influence each other. We detail and overcome the technical challenges that arise from considering interacting topics. We conduct a systematic evaluation of MPDHP on a range of synthetic datasets to define its application domain and limitations. Finally, we develop a use case of the MPDHP on Reddit data. At the end of this article, the interested reader will know how and when to use MPDHP, and when not to.

Keywords: Spreading process · Network Inference · Clustering · Bayesian Nonparametrics

1 Pseudo algorithm

The goal of the SMC algorithm illustrated in Fig. 1 is to jointly infer textual documents' clusters and the dynamics associated with them. The references in the figure correspond to the equations numbering of the main article. The algorithm runs as follows. First, the algorithm computes each cluster's posterior probability for a new observation by multiplying the temporal prior on cluster allocation with the textual likelihood. It results in an array of $K+1$ probabilities, where K is the number of non-empty clusters. A cluster label is then sampled from this probability vector. If the empty $(K+1)^{th}$ cluster is chosen, the new observation is added to this cluster, and its dynamics are randomly initialized

(i.e. a $(K+1)^{th}$ row and a $(K+1)^{th}$ column are added to the parameters matrix α). If a non-empty cluster is chosen, its dynamics are updated by maximizing the new likelihood. The process then goes on to the next observation.

This routine is repeated N_{part} times in parallel. Each parallel run is referred to as a *particle*. Each particle keeps track of a series of cluster allocation hypotheses. After an observation has been treated, we compute the particles likelihood given their respective cluster allocations hypotheses. Particles that have a likelihood relative to the other particles' one below a given threshold ω_{thres} are discarded and replaced by a more plausible existing particle.

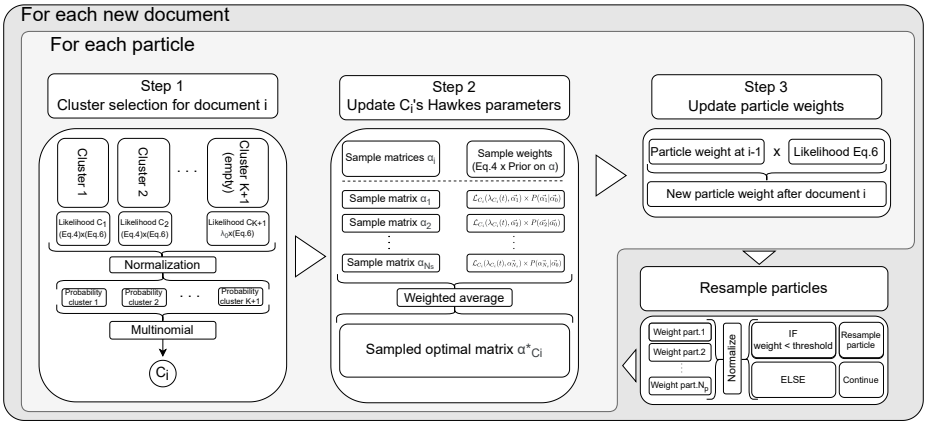

Fig. 1. Schematic workflow of the SMC algorithm — For each new observation from a stream of document, we run steps 1 (sample document's cluster), 2 (update sampled cluster's internal dynamics) and 3 (update particle likeliness) for each particle, and then discard particles containing the less likely hypothesis on cluster allocation.

2 Uninformative textual content and entangled dynamics

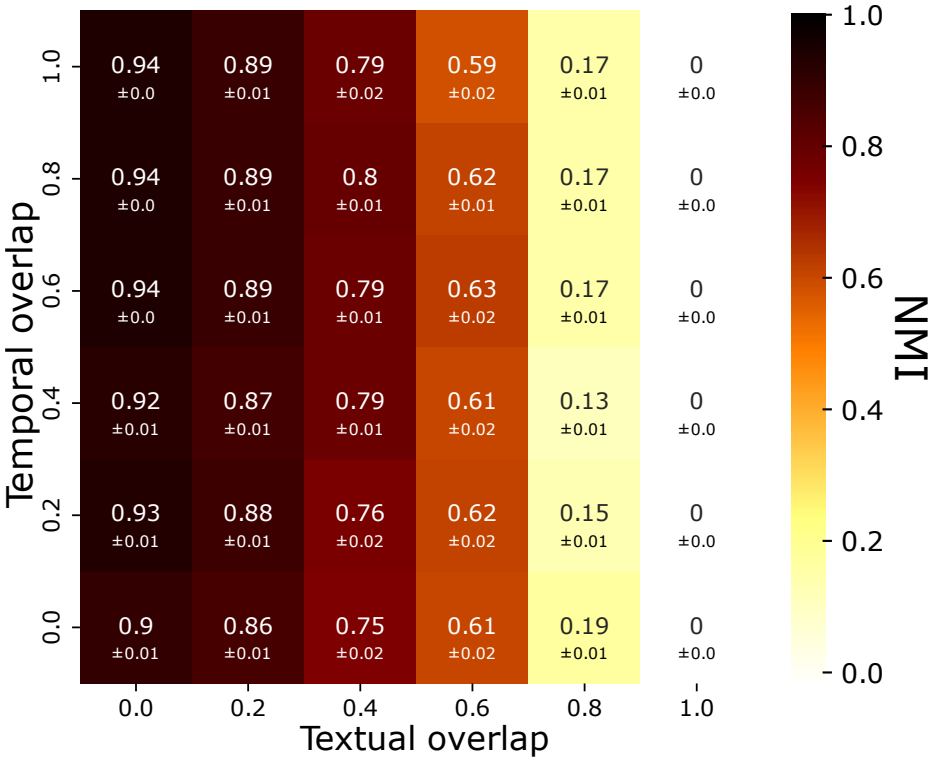

Fig. 2. MPDHP handles scarce textual or temporal information — MPDHP handles challenging cases provided either textual of temporal information is informative enough (temporal overlap of 0 and textual overlap of 0.7; temporal overlap of 1 and textual overlap of 0.4), and fails when both are uninformative (overlaps of 1).

In Fig. 2, we plot the results of MPDHP for different values of textual and temporal overlap. Textual overlap is defined as in the main article. The influence kernel of cluster c' on cluster c can be written $\alpha_{c,c'} \cdot \kappa(t)$. For each cluster c, we generate values of $\alpha_{c,c'}$ so that the overlap between all the functions in the set $\{\alpha_{c,c'} \cdot \kappa(t)\}_{c'}$ equals a given value. The idea is to test whether MPDHP is robust when clusters have similar dynamics.

Overall, we see that when the textual overlap is small, MPDHP yields good results independently from the temporal overlap. It means that in this case, the textual content is enough to differentiate clusters despite their dynamics being similar. However, as textual content gets less informative (textual overlap $\geq$ 0.6), results are better when the temporal overlap is low. In these cases, textual information is not enough and MPDHP relies more on temporal data. Overall,

MPDHP handles challenging cases provided either textual of temporal information is informative enough – for instance temporal overlap of 0 and textual overlap of 0.7, or temporal overlap of 1 and textual overlap of 0.4. It fails when both are uninformative – for instance, temporal and textual overlaps of 1.

3 Computational needs

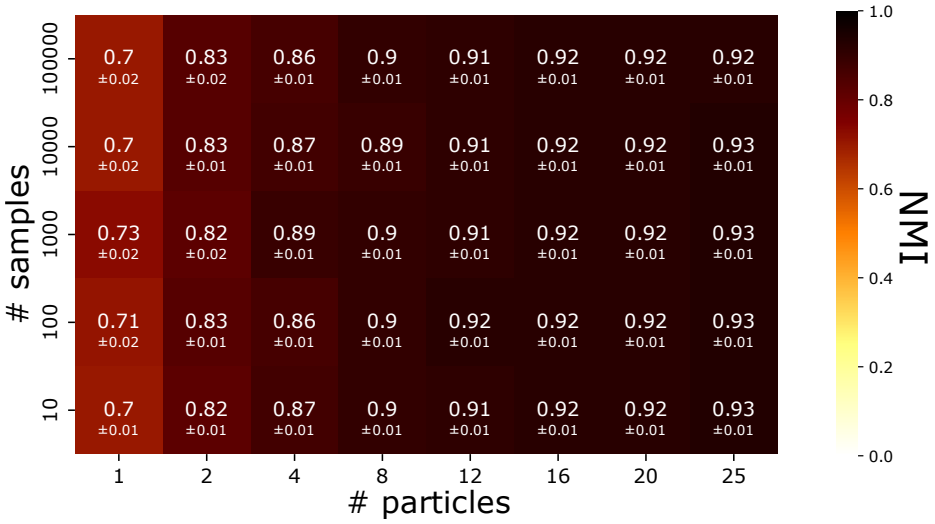

Fig. 3. How complex should the algorithm be — Performance of MPDHP using different versions of the Sequential Monte-Carlo algorithm. Here, we plot the model's performance with respect to the number of sample matrices used to estimate the kernel's weights α_c and the number of particles N_{part} used for the inference. Overall, MPDHP functions well with few computational resources.

We estimate how much computational resources we must allocate to MPDHP's sequential Monte-Carlo (SMC) inference algorithm in order to obtain good results. In Fig. 3, we plot the model's performance against the two main optimization parameters –the number of samples matrices and the number of particles. We recall that the samples matrices are used to infer each value of the kernel weights matrices for cluster c, noted α_c; the more sample matrices, the better the estimation. The number of particles represent the number of different cluster allocations hypothesis explored by the SMC algorithm at each step; the more particles, the more hypotheses are tested simultaneously. Overall, we see that MPDHP works well with few resources. In our experiments, results do not seem to improve significantly when using more than 20 particles, and when using more than 1000 sample vectors.

4 On the temporal concentration parameter λ_0

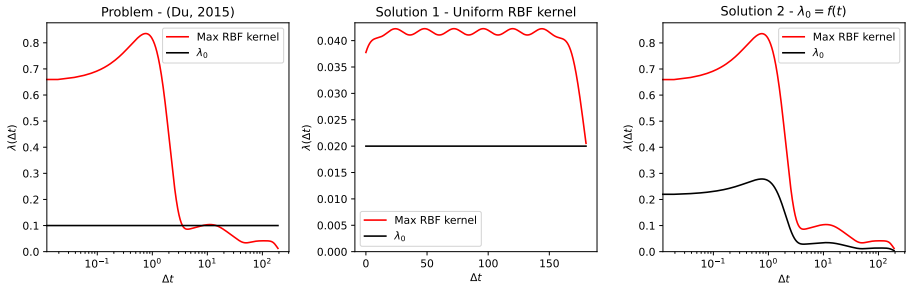

Fig. 4. Choosing the right temporal concentration parameter λ_0 — The choice of the temporal concentration parameter λ_0 can lead to bias. **(Left)** The problem with its choice in [1] is that events happening at large time ranges are likely to go undetected, as the Hawkes intensity at these ranges cannot be larger than λ_0. **(Middle)** A first solution consists in paving the space with evenly spaced Gaussian functions that all share the same standard deviation. **(Right)** A second solution is to make λ_0 a function of time so that its ratio with the temporal kernel remains constant.

While not specifically related to the implementation of the multivariate case, we discuss in this paragraph an important consideration when designing DHP-based models. In most of recently published works on the topic [1–3], inference on real-world processes is done using a RBF temporal kernel. It means that time is paved with Gaussian functions centered at various points in time; the parameter α in DHP-based models accounts for the weights given to each of these Gaussian functions.

In these works, the kernel is chosen so that it accounts for different time scales by centering Gaussian functions on unevenly spaced points in time. The standard deviation of each of these entries vary to account for larger time ranges. However, all values of a Gaussian function are small when their standard deviation is large, for normalization reasons –the maximum value of a Gaussian function whose standard deviation is σ is $\frac{1}{\sqrt{2\pi\sigma^2}}$.

In the SMC algorithm, this RBF kernel is evaluated at a single point in time, and confronted to the temporal concentration parameter λ_0 to determine whether to open a new cluster. In [1, 3], such values are compared to λ_0 constant in time. It means that, mechanically, these methods cannot detect observations triggered by such Gaussian functions as their value is systematically lower than λ_0 –typically at long time ranges in [1, 3], which can be seen from these articles' kernel plots that fade as time goes. We illustrate the problem in Fig. 4 (left).

Consider for instance the RBF kernel used in [1, 3], with the Gaussian means equal to 0.5, 1, 8, 12, 24, 48, 72, 96, 120, 144 and 168 hours, and the corresponding deviations equal to 1, 1, 8, 12, 12, 24, 24, 24, 24, 24, and 24 hours. The authors

used $\lambda_0 = 0.01$. For the last entry of their RBF kernel, the maximum value of the Gaussian function $\mathcal{G}(\mu = 168; \sigma = 24)$ is about 3.10^{-4}, which is much smaller than $\lambda_0 = 1.10^{-2}$. It means that even for a cluster whose intensity function only acts at long-ranges, the chances of spotting events triggered by such clusters are about 3%. This makes the models presented in [1, 3] unfit to spot long-range interactions.

There are two ways to overcome this problem (that we illustrate in Fig. 4 middle and right), so that λ_0 can be consistently confronted to the clusters' temporal kernels:

- To consider an RBF kernel whose Gaussian function all share the same deviation, while keeping λ_0 constant. We choose this solution in the follow-up experimental section.
- To consider a λ_0 that can vary in time according to the maximum value of the RBF kernel at different time points –which depends on their standard deviation.

5 Reddit dataset characteristics

We present some characteristics of the dataset used for the real-world experiments of the main article in Fig. 5.

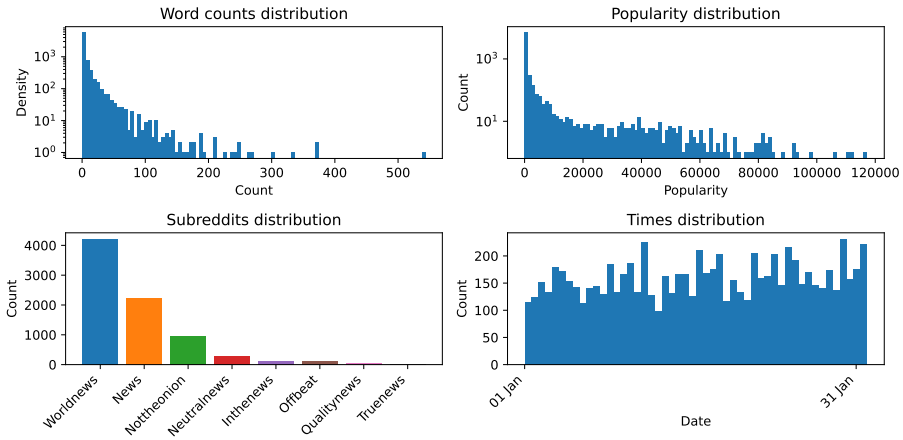

Fig. 5. Characteristics of the Reddit News dataset — For ∼8,000 headlines and ∼7,500 different words (Top-Left) Distribution of the words count (Top-Right) Distribution of headlines popularity (Bottom-Left) How many headlines per subreddit (Bottom-Right) How publications spread over time

References

1. Du, N., Farajtabar, M., Ahmed, A., Smola, A., Song, L.: Dirichlet-hawkes processes with applications to clustering continuous-time document streams. 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2015)
2. Mavroforakis, C., Valera, I., Gomez-Rodriguez, M.: Modeling the dynamics of learning activity on the web. In: Proceedings of the 26th International Conference on World Wide Web. p. 1421–1430. WWW '17 (2017)
3. Poux-Médard, G., Velcin, J., Loudcher, S.: Powered hawkes-dirichlet process: Challenging textual clustering using a flexible temporal prior. ICDM (2021)