REVISITING "SHALLOW" MUSIC REPRESENTATION LEARNING WITH HDPGMM

First Author
Affiliation1
author1@ismir.edu

Second Author
Retain these fake authors in submission to preserve the formatting

Third Author
Affiliation3
author3@ismir.edu

ABSTRACT

The abstract should be placed at the top left column and should contain about 150-200 words.

1. INTRODUCTION

Deep Learning (DL) on music signal has been a major key methodological shift in previous decade in the field of Music Information Retrieval (MIR). One of the core premises of DL is that it can learn useful feature, or representation of the input music audio signal [1,2]. Thanks to the diverse architectures and layers are invented, this "automatic" feature learning can handle wide range of data structure that are common in music domain (i.e., audio signal, lyrics, graph, etc.). Such flexibility and expressiveness is deemed as the key to its vast success.

In the pre-deep learning era, music representation learning still was discussed with "shallower" class of models [2–4].

2. HDPGMM

In this subsection, we introduce the main model we employ for this study; the Hiararchical Dirichlet Process Gaussian Mixture Model (HDPGMM) [5,6]. To introduce the model properly it would be useful to start discussing the Dirichlet Process Mixture Model (DPMM), the model on which HDPGMM is extended.

2.1 Dirichlet Process Mixture Models

Mixture models such as GMMs assume a finite number of components from which each of observed feature vectors are drawn. It is well known that finding the optimal number K of the mixuture components is a difficult problem. There are a few approaches that can be useful to estimate K such as using the cross-validation. Dirichlet Process Mixture Model (DPMM) circumbents this problem by parametrizing the number of mixtures as part of the model. DP plays a central role for such models.

DP is a stochastic process which draws random probability distributions. Due to this property, it is often described as a distribution over distributions [7]. It can be also seen as the infinite dimensional generalization of the Dirichlet distributions [7]. This aspect is the core building block of the Bayesian non-parametric models such as infinite mixture models. Setting DP as a prior distribution for the responsibility π (also often referred as mixture probability) of mixture components allows the mixture model to infer both the relative weight among components and the appropriate number of maximum components for given observations.

Among a several ways to represent DP, we introduce the stick-breaking construction [8]. [1] Stick-breaking constructs DP in a simple and general manner. Formally, it is as follows:

$$\begin{aligned} \beta'_k &\sim \text{Beta}(1,\gamma) & \phi_k &\sim H \\ \beta_k &= \beta'_k \prod_{l=1}^{k-1}(1-\beta'_l) & G_0 &= \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k} \end{aligned} \tag{1}$$

where two euqations in the left column represent the draw of infinite dimensional weight β_k which sums to one. Notably, the distribution for β is also referred as $\beta \sim \text{GEM}(\gamma)$ [9]. In the right column, H denotes the base distribution from which variable ϕ_k lying in some space Φ is drawn. The right bottom equation defines the draw of the probability measure G_0, where δ_{ϕ_k} means the point mass centered at the component ϕ_k. Altogether, Eq. 1 constructs the DP $G_0 \sim \text{DP}(\gamma, H)$. Figure 1 depicts the process in graphical way. In mixture model context, we want to infer mixture components $\{\phi_1, \phi_2, \cdots, \phi_k, \cdots\}$ that fits to the data observations $\{x_1, x_2, \cdots, x_n\}$, which are assumed to be drawn from distributions $F(\phi)$ parameterized with variable ϕ (i.e., mean and covariance $\phi = \{\mu, \Sigma\}$ in case of the multivariate Gaussian F). we can now use DP to draw ϕ as ith mixture components corresponds to the ith observation x_i by introducing the cluster assignment variable $y_i \sim \text{Mult}(\beta)$:

$$\begin{aligned} \beta|\gamma &\sim \text{GEM}(\gamma) & \phi_k|H &\sim H \\ y_i|\beta &\sim \text{Mult}(\beta) & x_i|y_i, \{\phi_k\} &\sim F(\phi_{y_i}) \end{aligned} \tag{2}$$

[1] Literature commonly chooses the Chinese Restaurant Process (CRP) for a illustrative metaphor for DP as it is intuitive and well explains various properties of DP [7]. We mainly discuss DP with the stick-breaking construction due to its further usage in the model inference within the work. For readers interested in other metaphorical descriptions of DP, we kindly refer them to [5, 7]

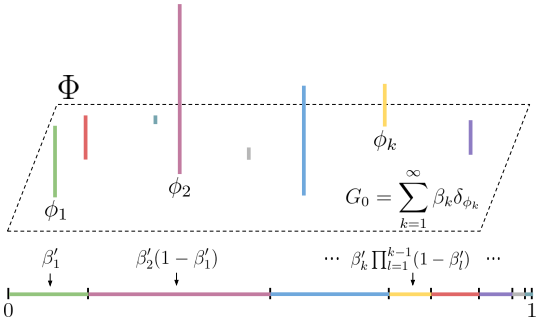

Figure 1. Illustration of stick-breaking construction in Eq. 1

where ϕ_{y_i} denotes the component parameter ϕ indexed by assignment variable y_i corresponding to the ith observation x_i.

2.2 Hierarchical DPMM

In many data structure, groupings of atomic data points arise naturally (i.e., audio frames within a song, songs from an artist, words of a lyrics). Hierarchical DP (HDP) is an extention of DP modelling "groupings" by imposing group-level DPs derived from the "global-level" or "corpus-level" DP as the global pool of components [6]. Following Sethuraman's stick-breaking construction [10], jth group-level DP can be expressed as follows:

$$\begin{aligned} \pi'_{jt} &\sim \text{Beta}(1, \alpha_0) & \psi_{jt} &\sim G_0 \\ \pi_{jt} &= \pi'_{jt} \prod_{l=1}^{t-1}(1-\pi'_{jl}) & G_j &= \sum_{t=1}^{\infty} \pi_{jt} \delta_{\psi_{jt}} \end{aligned} \tag{3}$$

As seen above, HDP indeed appears as the recursion of multiple levels of DPs [2] . Notably, the base distribution G_0 of each group-level DP is from the corpus-level DP. This relationship allows to map group-level atoms ψ_{jt} to the corpus-level atoms ϕ_k. Wang et al. introduce a series of indicator variables c_{jt} which maps ψ_{jt} and ϕ_k as follows [10]:

$$\begin{aligned} c_{jt} &\sim \text{Mult}(\beta) \\ \psi_{jt} &= \phi_{c_{jt}} \end{aligned} \tag{4}$$

where β is drawn from the corpus-level DP in Eq. 1. It simplifies the model as we do not need to explicitly represent ψ_{jt} [10]. Finally, we can represent HDPMM by introducing another indicator variable $z_{jn} \sim \text{Mult}(\pi_j)$ for nth observation x_{jn} within the jth group, similarly to Eq. 2:

$$\begin{aligned} \pi_j|\alpha_0 &\sim \text{GEM}(\alpha_0) & \theta_{jn} &= \psi_{jz_{jn}} = \phi_{c_{jz_{jn}}} \\ z_{jn}|\pi_j &\sim \text{Mult}(\pi_j) & x_{jn}|z_{jn}, c_{jt}, \{\phi_k\} &\sim F(\theta_{jn}) \end{aligned} \tag{5}$$

where we use the indicator z_{jn} to select ψ_{jt}, which eventually is mapped as $\phi_{c_{jz_{jn}}}$ that represents the parameter θ_{jn} to draw the observation x_{jn}. HDPGMM is then defined by simply setting F as the (multivariate) Gaussian distribution and H as one of distribution from which we can sample the mean and covariance (i.e., Gaussian-inverse Wishart distribution). Figure 2 depicts the HDPGMM graphically.

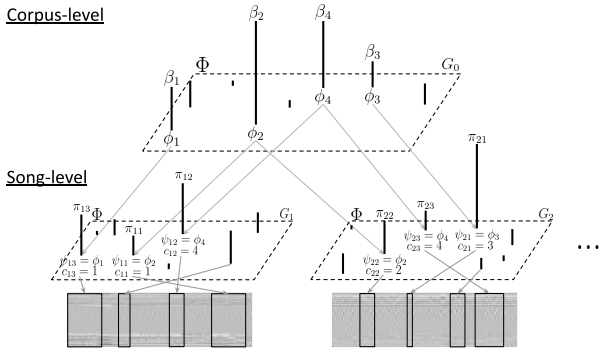

Figure 2. Illustration of 2-level HDPMM within corpus-song context. The top illustration depicts the corpus-level DP similar to Figure 1. The second row illustrations describe the draw of second level (song-level) DPs per song from the corpus-level DP. DPMM assumes that song features are drawn from each song-level DPs as depicted in the third rows of the image.

2.3 Inference Algorithm

In this section we discuss the inference (training) algorithm. We employ the online Variational Inference (VI) [10]. VI is one of the common choices to infer a fully Bayesian models and usually significantly faster than other methods such as Markov Chain Monte Carlo (MCMC) with the expense of its relative precision. VI seeks the simpler, approximated version of the true posterior by minimizing the Kullback-Leibler (KL) divergence between approximation $q(Z)$ and the true posterior $p(Z|X)$, where Z denote the set of latent variables and parameters that we want to find, and X refers a set of observations [11]. One of the popular simpliciation is full-factorization of the distribution $q(Z) = \prod_{i=1}^{|Z|} q_i(Z_i)$. In the context of HDPGMM, we have the following factorization:

$$q(\beta', \pi', c, z, \phi) = q(\beta')q(\pi')q(c)q(z)q(\phi) \tag{6}$$

where β', π', c, z denote the corpus-level and group-level stick proportion, group-level component selection variable, and finally the observation-level component selection variable, respectively. ϕ refers the parameter(s) for the F which draws the atomic observation, which is set as (multivariate) Gaussian in our context. Thus, ϕ includes the means μ and precision matrices Λ of each Gaussian component [3] .

Each variational distributions further factorize as follows:

$$\begin{aligned} q(\beta') &= \textstyle\prod_k^{K-1} \text{Beta}(\beta'_k|u_k, v_k) \\ q(\pi') &= \textstyle\prod_t^{T-1} \text{Beta}(\pi'_t|a_t, b_t) \\ q(c) &= \textstyle\prod_j \prod_t \text{Mult}(c_{jt}|\varphi_{jt}) \\ q(z) &= \textstyle\prod_j \prod_n \text{Mult}(z_{jn}|\zeta_{jn}) \end{aligned} \tag{7}$$

[2] It implies naturally that multiple levels are possible (i.e., corpus - author - document), if it suits to the data structure.

[3] We adopt the result from [11], where the Gaussian-Wishart distribution is used for the prior.

where u_k, v_k, a_t, b_t denote the variational parameters for the Beta distributions for corpus-level and document-level stick proportions, respectively. $\varphi_{jt} \in \mathbb{R}^K, \zeta_{jn} \in \mathbb{R}^T$ are the variational parameters for the Multinomial distribution to draw the selector c and z.

Notably, we truncate the inifinite Beta distributions by K and T, which is a common method for applying VI on the infinite mixture model. With sufficiently large number for the truncation, the model will still not be limited to the truncation, and will only use the number of components that suits for given dataset. Final variational distribution is the Gaussian-Wishart prior distribution which draws the Gaussian parameters $\phi = \{\mu, \Lambda\}$ for distribution F:

$$q(\phi) = \prod_k^K \mathcal{N}(\mu_k | m_k, (\lambda_k \Lambda_k)^{-1}) \mathcal{W}(\Lambda_k | W_k, \nu_k) \quad (8)$$

where we draw the precision $\Lambda_k \in \mathbb{R}^{d \times d}$ from Wishart distribution with the variational parameter $W_k \in \mathbb{R}^{d \times d}$ and $\nu_k \in \mathbb{R}$, and the mean $\mu_k \in \mathbb{R}^d$ is drawn by the precision weighted by $\lambda_k \in \mathbb{R}$ and mean $m_k \in \mathbb{R}^d$.

We then can obtain the optimal model by maximizing the lowerbound of the marginal log likelihood $\log p(X|Z)$ [11–13]:

$$\begin{aligned} \log p(X|Z) &\geq \mathbb{E}_q[\log p(X, \beta', \pi', c, z, \phi)] + H(q) \\ &= \sum_j \left\{ \mathbb{E}_q[\log (p(X_j|c_j, z_j, \phi) p(c_j|\beta') p(z_j|\pi'_j) p(\pi'_j|\alpha_0))] \right. \\ &\quad \left. + H(q(c_j)) + H(q(z_j)) + H(q(\pi'_j)) \right\} \\ &\quad + \mathbb{E}_q[\log p(\beta') p(\phi)] + H(q(\beta')) + H(q(\phi)) \end{aligned} \quad (9)$$

where $H(\cdot)$ denotes the entropy of given distribution, and $X_j = \{x_{j1}, x_{j2}, \cdots, x_{jN_j}\}$ is a set of observations within jth group. [4]

Using the standard result of VI [10, 11, 13], the update rules for group-level parameters are dreived as follows:

$$a_{jt} = 1 + \sum_n \zeta_{jnt} \quad (10)$$

$$b_{jt} = \alpha_0 + \sum_n \sum_{s=t+1}^T \zeta_{jnt} \quad (11)$$

$$\varphi_{jtk} \propto \exp(\sum_n \zeta_{jnt} \mathbb{E}_q[\log p(x_{jn}|\phi_k)] + \mathbb{E}_q[\log \beta_k]) \quad (12)$$

$$\zeta_{jnt} \propto \exp(\sum_{k=1}^K \varphi_{jtk} \mathbb{E}_q[\log p(x_{jn}|\phi_k)] + \mathbb{E}_q[\log \pi_{jt}]) \quad (13)$$

Similarly, the update rules for the corpus-level parameters are as follows [11]:

$$u_k = 1 + \sum_j \sum_{t=1}^T \varphi_{jtk} \quad (14)$$

$$v_k = \gamma + \sum_j \sum_t^T \sum_{l=k+1}^K \varphi_{jtl} \quad (15)$$

$$\lambda_k = \lambda_0 + N_k \quad (16)$$

$$m_k = \lambda_k^{-1}(\lambda_0 m_0 + N_k \bar{x}_k) \quad (17)$$

$$W_k^{-1} = W_0^{-1} + N_k S_k + \frac{\lambda_0 N_k}{\lambda_0 + N_k}(\bar{x}_k - m_0)(\bar{x}_k - m_0)^\intercal \quad (18)$$

$$\nu_k = \nu_0 + N_k \quad (19)$$

[4] Thus, the terms inside the sum over the group would further factorized into sums of these per-group observation. We omit them to avoid the equation being too crammed.

where $\lambda_0 \in \mathbb{R}, m_0 \in \mathbb{R}^d, \nu_0 \in \mathbb{R}, W_0 \in \mathbb{R}^{d \times d}$ are the hyperparameters corresponding to the weight, location, degrees of freedom, and scale of Gaussian-Wishart distribution. The sufficient statistics and expectations used above update rules are defined as follows:

$$\begin{aligned} &\mathbb{E}_q[\log \beta_k] = \mathbb{E}_q[\log \beta'_k] + \textstyle\sum_{l=1}^{k-1} \mathbb{E}_q[\log (1 - \beta'_l)] \\ &\mathbb{E}_q[\log \beta'_k] = \Psi(u_k) - \Psi(u_k + v_k) \\ &\mathbb{E}_q[\log (1 - \beta'_k)] = \Psi(v_k) - \Psi(u_k + v_k) \\ &\mathbb{E}_q[\log \pi_{jt}] = \mathbb{E}_q[\log \pi'_{jt}] + \textstyle\sum_{s=1}^{t-1} \mathbb{E}_q[\log (1 - \pi'_s)] \\ &\mathbb{E}_q[\log \pi'_{jt}] = \Psi(a_{jt}) - \Psi(a_{jt} + b_{jt}) \\ &\mathbb{E}_q[\log (1 - \pi'_{jt})] = \Psi(b_{jt}) - \Psi(a_{jt} + b_{jt}) \\ &N_k = \textstyle\sum_j \sum_n r_{jnk} \\ &\bar{x}_k = \frac{1}{N_k} \textstyle\sum_j \sum_n r_{jnk} x_{jn} \\ &S_k = \frac{1}{N_k} \textstyle\sum_j \sum_n r_{jnk} (x_{jn} - \bar{x}_k)(x_{jn} - \bar{x}_k)^\intercal \end{aligned}$$

where $\Psi(\cdot)$ refers the digamma function, and $r_{jnk} = \sum_{t=1}^T \zeta_{jnt} \varphi_{jtk}$ is the inferred responsibility of nth observation of jth group on kth component. Standard batch update would compute statistics across the entire corpus, and updates the corpus level parameters. However, it may be slow or may suffer by too large or small numbers accumulated from a large scale corpus.

In this work, we employ the online VI where corpus-level parameters are updated per every group or mini-batch of group passed. In this way, the early phase of model inference can be accelerated substantially compared to the full-batch update [10, 14]. The corpus level update then contoled by the learning rate $\rho_t = (\tau_0 + t)^{-\kappa}$, which is decayed over iterations controlled by the offset parameter $\tau_0 > 0$ and scale $\kappa \in (0.5, 1]$:

$$Z_t = (1 - \rho_t) Z_{t-1} + \rho_t \tilde{Z}_t \quad (20)$$

where $\tilde{Z}_t$ means one of the corpus-level parameter in Eq. (14) to (19) updated by the given mini-batch at tth iteration, while Z_{t-1} is the current parameter. To properly scale the update with respect to the mini-batch size, we weight them by the factor of $w = \frac{|\tilde{X}|}{|X|}$, where $|X|, |\tilde{X}|$ denote the number of groups within the entire observation set X and the mini-batch of groups $\tilde{X}$.

Combining altogether, the overall inference algorithm is described in Algorithm 1.

2.4 Further Regularization

Inspired by the implementation of [10, 13], we apply the further regularization on the model. Specifically, we "splash" the uniform noise to the inferred responsibility r_{jn} as it can be biased if the groups are corrupted or impcomplete, such as the preview audio of the entire song:

$$\tilde{r}_{jn} = (1 - \eta_t) r_{jn} + \eta_t e \quad (21)$$

η_t is the regularization coefficient that determines the extent uniform noise $e = (e_k)_{k=1}^K$ is mixed into r_{jn}. $\eta_t = \frac{\eta_0 * 10^3}{(t + 10^3)}$ also is defined as decaying function similar to the learning rate ρ_t.

Algorithm 1: Online VI for HDPGMM

Initialize $\phi = (\phi_k)_{k=1}^K$, $u = (u_k)_{k=1}^K$, $v = (v_k)_{k=1}^K$ randomly. Set $t = 1$.

while *Stopping critrion is not met* **do**

  Fetch a random mini-batch of groups $\tilde{X}$

  repeat

    Update a_j, b_j, ζ_jandφ_j using Eq. (10) to (13)

  until *mini-batch likelihood stops improving*;

  Compute $u_k, v_k, \lambda_k, m_k, W_k$, and$\nu_k$ using Eq. (14) to (19)

  Set $\rho_t = (\tau_0 + t)^{-\kappa}$, $t \leftarrow t + 1$

  Update $u_k, v_k, \lambda_k, m_k, W_k$, and$\nu_k$ using Eq. 20

end

2.5 DPMMs in MIR

3. EXPERIMENTAL SETUP

4. RESULT AND DISCUSSION

5. CONCLUSION AND FUTURE WORKS

6. REFERENCES

[1] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Moving beyond feature design: Deep architectures and automatic feature learning in music informatics," in *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*, F. Gouyon, P. Herrera, L. G. Martins, and M. Müller, Eds. FEUP Edições, 2012, pp. 403–408. [Online]. Available: http://ismir2012.ismir.net/event/papers/403-ismir-2012.pdf

[2] P. Hamel, M. E. P. Davies, K. Yoshii, and M. Goto, "Transfer learning in mir: Sharing learned latent representations for music audio classification and similarity," in *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013, Curitiba, Brazil, November 4-8, 2013*, A. de Souza Britto Jr., F. Gouyon, and S. Dixon, Eds., 2013, pp. 9–14. [Online]. Available: http://www.ppgia.pucpr.br/ismir2013/wp-content/uploads/2013/09/76_Paper.pdf

[3] J. Wang, H. Lee, H. Wang, and S. Jeng, "Learning the similarity of audio music in bag-of-frames representation from tagged music data," in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*, A. Klapuri and C. Leider, Eds. University of Miami, 2011, pp. 85–90. [Online]. Available: http://ismir2011.ismir.net/papers/PS1-8.pdf

[4] J. Nam, J. Herrera, M. Slaney, and J. O. S. III, "Learning sparse feature representations for music annotation and retrieval," in *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*, F. Gouyon, P. Herrera, L. G. Martins, and M. Müller, Eds. FEUP Edições, 2012, pp. 565–570. [Online]. Available: http://ismir2012.ismir.net/event/papers/565-ismir-2012.pdf

[5] M. D. Hoffman, D. M. Blei, and P. R. Cook, "Content-based musical similarity computation using the hierarchical dirichlet process," in *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*, J. P. Bello, E. Chew, and D. Turnbull, Eds., 2008, pp. 349–354. [Online]. Available: http://ismir2008.ismir.net/papers/ISMIR2008_130.pdf

[6] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006. [Online]. Available: https://doi.org/10.1198/016214506000000302

[7] Y. W. Teh, "Dirichlet process," in *Encyclopedia of Machine Learning and Data Mining*, C. Sammut and G. I. Webb, Eds. Springer, 2017, pp. 361–370. [Online]. Available: https://doi.org/10.1007/978-1-4899-7687-1_219

[8] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[9] J. Pitman, "Poisson-dirichlet and gem invariant distributions for split-and-merge transformations of an interval partition," *Comb. Probab. Comput.*, vol. 11, no. 5, pp. 501–514, 2002. [Online]. Available: https://doi.org/10.1017/S0963548302005163

[10] C. Wang, J. W. Paisley, and D. M. Blei, "Online variational inference for the hierarchical dirichlet process," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, ser. JMLR Proceedings, G. J. Gordon, D. B. Dunson, and M. Dudík, Eds., vol. 15. JMLR.org, 2011, pp. 752–760. [Online]. Available: http://proceedings.mlr.press/v15/wang11a/wang11a.pdf

[11] C. M. Bishop and N. M. Nasrabadi, "*Pattern Recognition and Machine Learning*," *J. Electronic Imaging*, vol. 16, no. 4, p. 049901, 2007. [Online]. Available: https://doi.org/10.1117/1.2819119

[12] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Mach. Learn.*, vol. 37, no. 2, pp. 183–233, 1999. [Online]. Available: https://doi.org/10.1023/A:1007665907178

[13] D. M. Blei and M. I. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Analysis*, vol. 1, no. 1, pp. 121 – 143, 2006. [Online]. Available: https://doi.org/10.1214/06-BA104

[14] M. D. Hoffman, D. M. Blei, and F. R. Bach, "Online learning for latent dirichlet allocation," in *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds. Curran Associates, Inc., 2010, pp. 856–864. [Online]. Available: https://proceedings.neurips.cc/paper/2010/hash/71f6278d140af599e06ad9bf1ba03cb0-Abstract.html