Applying Data Science for HCM in the Modern Enterprise: a journey, lessons learnt, and pitfalls

Data Science for Human Capital Management (DSHCM) Workshop, IEEE ICDM 2017

Moninder Singh

IBM Research AI, Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA

Human Capital Management (HCM) & Workforce Analytics at IBM: A Decade Long Journey

Over a decade or so ago, when work at IBM started on HCM and Workforce Analytics, focus was more business-centric.

- How will our workforce evolve in the future and which policies will lead to optimal workforce composition

  (Y. Lu, A. Radovanovic, and M. Squillante. “Workforce management and optimization using stochastic network models”. IEEE SOLI, 2006.)

- How can we measure true capacity of our human capital and analyze/improve productivity and revenue performance

  (M. Singh, D. Bhattacharjya, L. Deleris, D. Katz-Rogozhnikov, M. Squillante, others. “The growth and performance diagnostics initiative: A multidimensional framework for sales performance analysis and management”. Service Science, 2011.)

Human Capital Management (HCM) & Workforce Analytics at IBM: A Decade Long Journey

- How can we help organizations make better informed resource planning and allocation decisions

  (Y. Richter, Y. Naveh, D. L. Gresh, and D. P. Connors, “Optimatch: applying constraint programming to workforce management of highly skilled employees,” Int. J. Services Operations and Informatics, 2008)

- How can we compute demand forecasts, compute skill gap/gluts, and determine optimal capacity plans

  (Cao, J. Hu, C. Jiang, T. Kumar, T.-H. Li, Y. Liu, Y. Lu, S. Mahatma, A. Mojsilovic, M. Sharma, M. S. Squillante, and Y. Yu, “OnTheMark: Integrated stochastic resource planning of human capital supply chains”. Interfaces, 2011.)

Human Capital Management (HCM) & Workforce Analytics at IBM: A Decade Long Journey

With the rapidly changing needs of the marketplace (where demand for skills changes quickly), the focus then turned more human-capital centric

- How can we monitor employees at risk of leaving on a continual basis? Why are these employees at risk?

  (K. N. Ramamurthy, M. Singh, Y. Yu, J. Aspis, M. Iames, M. Peran and Q. Held. “A Talent Management Tool using Propensity to Leave Analytics”. IEEE DSAA, 2015. Best Application Paper.)

- Who are the high-risk employees that makes the most sense to try to retain and how?

  (M. Singh, K. R. Varshney, J. Wang, A. Mojsilovic, M. S. Squillante, Y. Lu, A. Gill, P. Faur, and R. Ezry. “An Analytics Approach for Proactively Combating Voluntary Attrition of Employees”. IEEE ICDM Workshop on Data Mining for Service, 2012)

Human Capital Management (HCM) & Workforce Analytics at IBM: A Decade Long Journey

- How can we identify expertise within our employees as new skills emerge?

  (K. R. Varshney, V. Chenthamarakshan, S. W. Fancher, J. Wang, D. Fang, and A. Mojsilovic. “Predicting Employee Expertise for Talent Management in the Enterprise”. KDD, 2014.)

- Who are the employees that are most suited to re-skill to new, in-demand skills?

  (K. N. Ramamurthy, M. Singh, M. Davis, J. A. Kevern, U. Klein and M. Peran. “Identifying employees for re-skilling using an analytics-based approach”. IEEE ICDM Workshop on Data Mining for Service, 2015)

  (M. Singh, K. Natesan Ramamurthy, and S. Vasudevan. ‘’ Propensity Modeling for Employee Re – Skilling’’. IEEE GlobalSIP, 2017.)

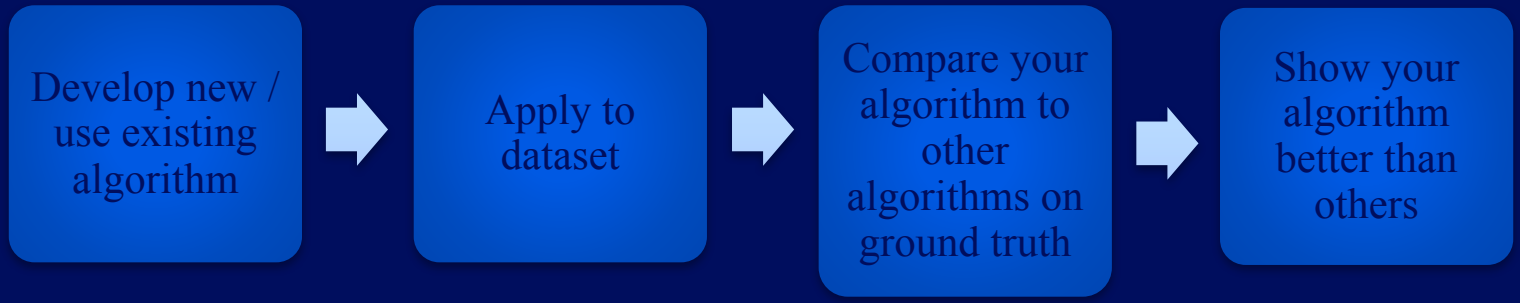
- How can we do better skill-capacity planning to meet future demands while satisfying current requirements (with skills getting obsolete)

  (S. Vasudevan, M. Singh, J. Mondal, M. Peran, B. Zweig, B. Johnston and R. Rosenfeld. “Estimating Fungibility between Skills by Combining Skill - Similarities Obtained from Multiple Data Sources”. IEEE ICDM Workshop on Data Science for Human Capital Management, 2017.

So What Did We Observe

- There is typically NO gold-standard (ground truth)
  - Solving new problem (e.g. which employee/skill can be best re-skilled) has no data against which model performance can be evaluated
  - Controlled experiments are not possible (costs/laws/policies etc)
- There is often lots of data/features
  - But that data is noisy/dirty
  - May be inconsistent/incorrect
  - Lots of features but mostly useless
- Underlying distributions often change over time
- End-user acceptance is very important
  - Interpretability and way of presentation of results is key

Contrast with common process



Too much focus on algorithm

Accept data/features as given

Little focus on domain/data understanding

What if ground-truth is not available

What if ground-truth is not really valid???

Too much focus on algorithm performance

Little focus on model interpretability, presentation of results, stability over time

So What Did We Learn

Study/know the domain

Understand the data

Do not blindly trust the data

Feature construction is important

So is Interpretability

Pay attention to Stability of Results over time

All these often go hand in hand

Study/know your domain

- Data is often collected as a part of the business process. It is not collected with modeling in mind.
  - Modeling is often an afterthought
  - E.g. HR data has long been collected in an enterprise; HR analytics is a recent phenomenon
- We typically throw various algorithms at the data provided to us
  - Too much focus on the algorithm
  - Little interest in understanding domain
  - How would a domain expert do the task?
  - Businesses collect/provide data they think is important for business, not analytics. How would you even know what to ask for without knowing the domain?

Study/know your domain

Example 1: Predicting risk of voluntary attrition of employees w.r.t Compensation

- 200K employees considered to evaluate attrition risk and possible retention bonus.
  - Threw various algorithms (Decision trees, regression, SVM, random forests, etc.) at data
  - Compensation had no bearing on attrition risk ... BIG SURPRISE
  - Employees were based in multiple countries/locations
    - Still did not matter
  - Compensation based on several factors – location, experience, job role, skills, etc.
  - Important to compare within peer groups ... Strong signal observed

- While intuitive in this case, shows importance of understanding domain

Study/know your domain

Example 2: Predicting risk of voluntary attrition of employees wrt Time since promotion

- Once again, surprisingly, feature based on time since last promotion did not matter
- As in the case of compensation,
  - Career speed/trajectory varies according to a lot of factors (location, business, job, skills, etc.)
  - So, important to create features that compare against peer groups

Study/know your domain

- A little digression...
- This is true for other domains (non HCM) as well

Example 1: Predicting future patient costs based on prior diagnoses/treatments/costs etc.

- Blindly applying algorithms to raw data ignores facts that acute events cost a lot but don't re-occur, while chronic events occur repeatedly with continuous costs.
- Similarly, certain diseases often lead to/co-exist to other diseases, thus leading to increased costs

Example 2: Predicting final teeth positions (after orthodontic treatment) based on initial teeth positions to suggest treatment plan to new/inexperienced doctors

- Directly predicting final positions based on initial positions produced bad results
- Orthodontists typically identify a handful of different types of cases, and treat them differently. Data was available but not provided since no-one knew to ask for it.

Understand your data – do not blindly trust what is given

- Just because a model performs well does not mean it is correct
- This is especially true in business where data may be fed into a corporate warehouse from many different geographical/functional upstream data repositories
  - Often leads to inconsistencies, and sometimes, erroneous data
  - Even if this is not the case, your understanding of the data may not reflect what it truly is

Understand your data – do not blindly trust what is given

- Example 1: While modeling attrition at IBM , a very strong predictor for attrition risk was “time since last salary increase”
  - Intuitively, it made sense. If one has not got an increase for a long time, s(he) should be at increased risk for leaving
  - However, careful examination of model (decision tree) showed the exact reverse: model predicted high attrition risk if in fact employee got recent raise
  - In fact, it was an artifact of data. Data consisted of active employees just prior to annual salary raise cycle; Attriters were employees who had left in the last year.
    - Thus, active employees had not had a raise for a year or more; most attriters had got a raise within a few months of leaving
  - Good model but incorrect

Understand your data – do not blindly trust what is given

- Example 2: While modeling attrition for a large financial multi-national, a very strong predictor for attrition risk in one geography was “change of manager within first 6 months”
  - Global executives were quick to suggest changes
  - However, we asked to talk with up-stream database owners and business leaders
    - Found that new hires were typically all assigned to the same organization till an appropriate permanent position was made.
    - A spurious link with attrition was detected due to an artifact of the data

Understand your data – do not blindly trust what is given

- Example 3: While modeling attrition for the same large financial multi-national, “time since last salary raise” was found to have no bearing to attrition risk
  - Executives believed otherwise (and intuitively it made sense)
  - Further exploration if data revealed an interesting factoid
    - The data being used for modeling was from the 2008-2010 period
    - The time coincided with the financial crisis and massive layoffs
    - Raise or no raise, few left as there was no place to go...

Understand your data – do not blindly trust what is given

- Example 4: While modeling employees that could be easily re-skilled to a variety of consultant job roles, a very strong indicator was found in the presence of the job role of “Strategy Consultant” in an employee’s job history
  - It seemed that someone with a Strategy Consultant background could be easily re-skilled to virtually any of the consultant job roles.
  - As such, the model had very good performance
  - Nevertheless, talking to the business folks in the strategy area revealed that all new hires were automatically assigned a job-role of Strategy Consultant till an appropriate job –role was assigned.
  - Once again, a good model, albeit incorrect

Interpretability is important

- To get end-user buy-in, interpretability of models is paramount
  - In HCM, could be due to legal reasons
  - Convincing users to make investments on basis of black-box predictions may not be feasible
  - Especially important if (financial) cost of mistake is large
  - May also help identify actions that can be taken based on model results
  - Real-world data, especially in HCM domain, is very noisy
    - Complex methods typically overfit; don’t give much better performance but seriously impact interpretability
    - Can often get similar gains by domain-knowledge based feature construction

Interpretability is important

- Presentation of results to end-user is also important
  - Example: Is 20% risk of attrition good or bad?
    - In US, where typical attrition is 5-6%, it is bad
    - In India, where typical attrition is 40%+, it is good
    - To an executive looking at attrition risk globally, raw risk numbers will give wrong impression (e.g. India – all red, US – all green)
    - Showing relative risk (high/medium/low) may be better
    - Even a typical manager has trouble dealing with raw numbers...should she take action for an employee with 20% risk or not?

So is model (results) stability over time

- In business, it is seldom a once and done activity
- Models are typically used on a continuous basis
- Will need to be re-learned periodically
  - Should the results change dramatically because underlying population changed and your algorithm is very sensitive to change?
  - Example: Risk of attrition again...
    - Should risk of attrition of an individual (or factors affecting the risk) change over time, even though nothing really changed for that individual? What changed was underlying population and its characteristics...
  - By using domain knowledge to model carefully as well as using ensembles/model averaging, this can be alleviated, though needs to be balanced against model interpretability

What do we take away from this?

- While the following is true for all professionals, it is especially meant for new practitioners and/or graduate students

  - Don’t just focus on the algorithms; focus instead on the problem to be solved
  - Understand the domain
    - Feature creation using domain knowledge will often give you performance boosts
  - Question the data
    - If something is too good to be true, question it.
    - Don’t just take a model learned from the data as-is

What do we take away from this?

- Remember, you may not have ground-truth to evaluate your model (or ability to carry out controlled experiments)
  - If you understand domain and data, and model it correctly, you will likely do as well as possible

- Pay attention to interpretability of models/results as well as stability over time
  - Research in these areas is as critical as research in developing new algorithms.