Theme 1: Artificial Intelligence (AI) in Capacity Management

Introduction

Thank you for joining the AI in Capacity webinar, being organized as part of the Radio – EY GDS Hackathon. In this session, we give you a quick overview about the theme, the expectations and answer some of the common questions that you may have.

Business overview

In the experience management domain, also known as resource management, we get a lot of work requests which we term as demands. We have a set of employees available with varied skills, proficiencies, competencies who can execute on these work requests or demands. In our current scenario, the demand-to-supply match is a manual activity or a manual task. It is one of the core work activities performed by the resource management team. Approximately 60% of our overall activities are around demand supply management. The manual effort makes it pretty cumbersome and
non-usable to a lot of our business teams.

We want to build a system where we can capture all the demands in a structured manner, capturing all the relevant demand heuristics, which are mentioned in the theme, such as experience, skills, competencies, location, and resource level that's needed, and then map it to our supply pool, which is also mapped on similar lines.

We get the best fit for each demand request, mapped by the percentage of fitment to that demand. We are also looking at the reverse of a demand-to-supply match, which is a supply-to-demand match.

So, if we have employees who do not have work or a project assigned to them, we can put their specifications in the demand-supply match tool and identify what open projects are available within the organization, where their skills competencies, etc. have a direct or an associated match. The objective is that the entire work allocation demand management process becomes more automated and system, tool dependent and less manual than what it is today.

From an organization’s perspective, we have multiple service lines and business units, performing different activities. While some activities can be unique to a particular service line or business unit, some are common.

Hence, one of the broader outcomes that we want to achieve with this kind of a tool, or a process, is to be able to optimally utilize our employees, based on their skills.

If there is a demand, and there is a supply option available anywhere within the organization, we explore that before hiring anybody externally from the market.

Implementation details

In this section, we will take you through the detailed steps that we're looking for in this theme.

1. We will be receiving the demand data in the form of an Excel sheet.
2. For the demand data that we have received, there will be assignment of weightages for the search factors, basically, the demand heuristics that were mentioned earlier, experience, skills and competency location, greed, rank, bench aging, service line.
3. Based on the demand that is available, we need to identify employees from the available pool and match with the incoming demand based on the search precedence given.

Based on the weightages that were assigned on the search factors, it is expected that a recommendation is given to the demand that has been received. There is a fitment percentage, that was mentioned previously, which will help us classify the best bet, best fit and stretched fit for the given demand. When we say fitment percentage, any recommendation will be termed as a best fit, when it has more than 85% of the fitment criteria being managed successfully. Anything between 70% and 85% will be categorized as a stretched fit and 60 to 70% will be categorized as the best bet for the given demand. There should also be an option to redo the weightage factors.

When the weightage percentage are redone, there should be recommendations popping up based on the redone weightage factors. So, these are the detailed steps that we are looking for in the AI in Capacity Management process.

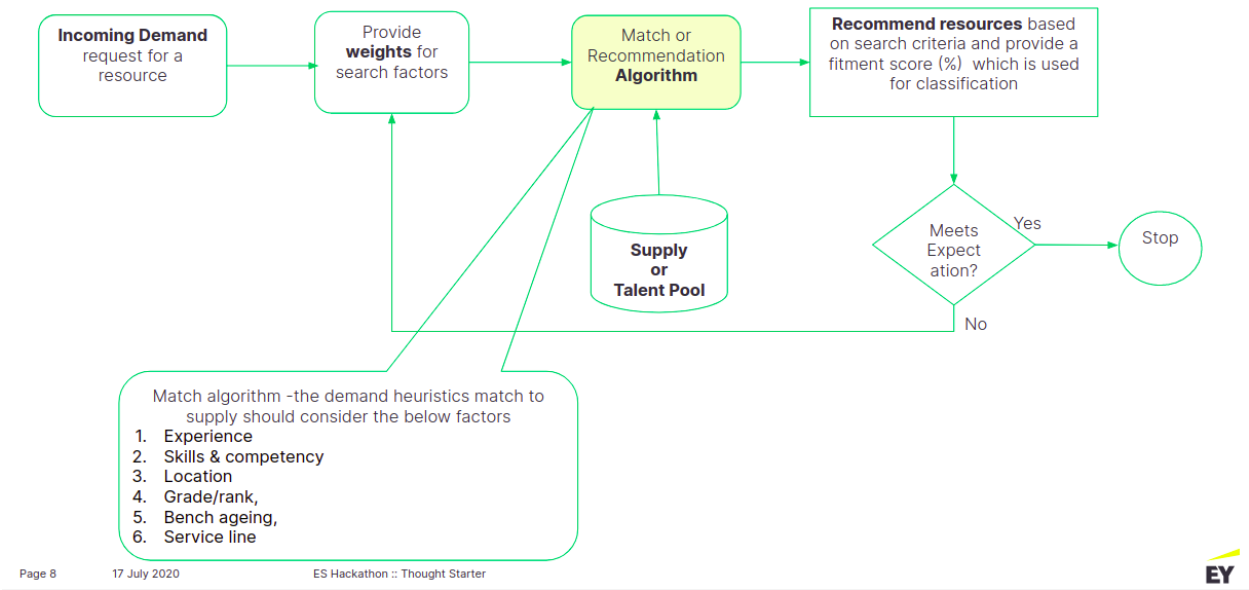
We would also like to discuss a little bit about the statistical goal and what precisely are we meaning to achieve through this hackathon.

The aim would be to build a recommendation engine, that would help in reducing the man hours that currently, the team is putting in while performing this activity manually. It is expected that the manual effort will reduce by 50%. Through this solution, we would also like to focus on improving the accuracy of the demand-supply match process by using the standard logic that has been put across.

The expected statistical solution is that the demand-supply matching process and the allocation of the employees will have three major segments, i.e., the best fit, stretched fit and the best bet. And the recommendation algorithm should also be considering the weightage ranking of factors, i.e., experience skills and competency location, grade, rank, bench aging, service line, etc.

We will now focus on the technical aspects of the solution.

Data flow



As explained earlier, you have an incoming request for an employee. It's called demand. An incoming demand would be coming in, and a user should be able to provide a search or search criteria on the set of things that is explained here, that's called as demand heuristics, i.e. experience, skills and competencies, location, rank, bench aging, etc.

So, essentially what we are looking for is to build a match or recommendation algorithm, which can understand the incoming demand, and is able to kind of match the right set of employees from the talent or supply data, then provide recommendations. We should have a feature where we could reiterate the search factor, so that we can redo this process as and when required.

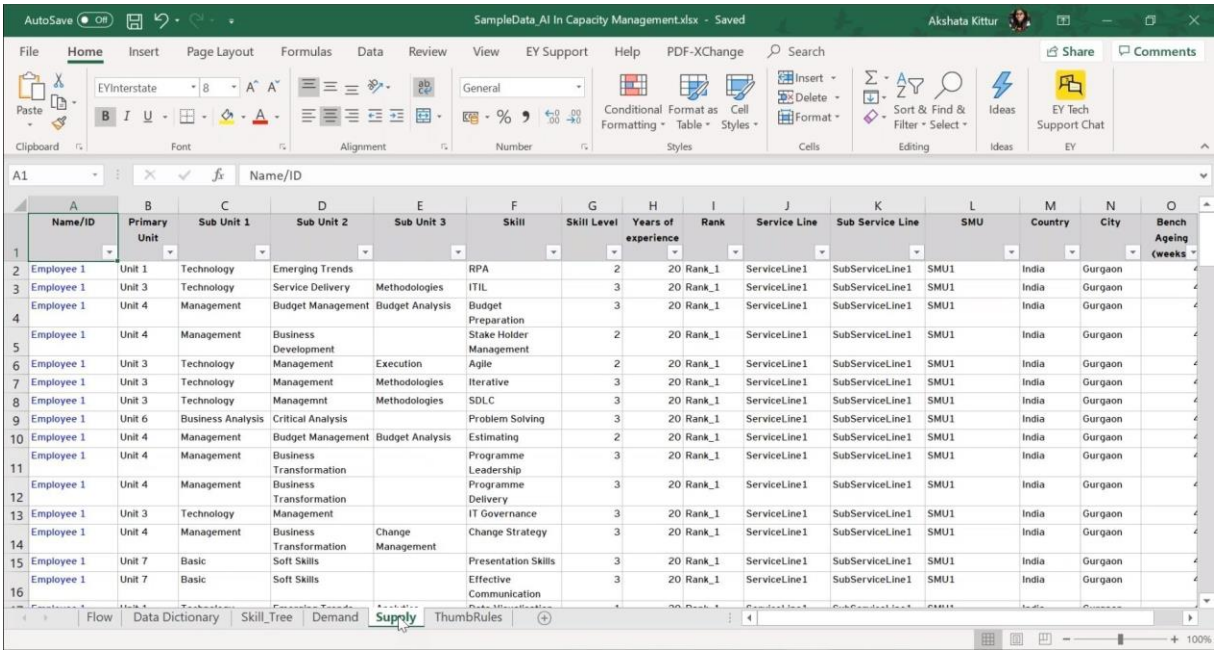
Dataset details

We'll just touch upon how the ***demand data*** would look like.

There is an Excel sheet which will contain a requester and then requestor ID, requestor service line, requestor sub-service line, requestor sub-management unit (SMU), that comprises of our internal department, structure or a business unit structure. We then have a job title, rank and the number of employee sources that acquired the country or location.

And then we'll move on to the skills, which would essentially mean primary skills, technical skills, and process skills, followed by years of experience.

Whatever we have at the moment is like a sample of data. Now, we'll touch upon the ***supply data***:



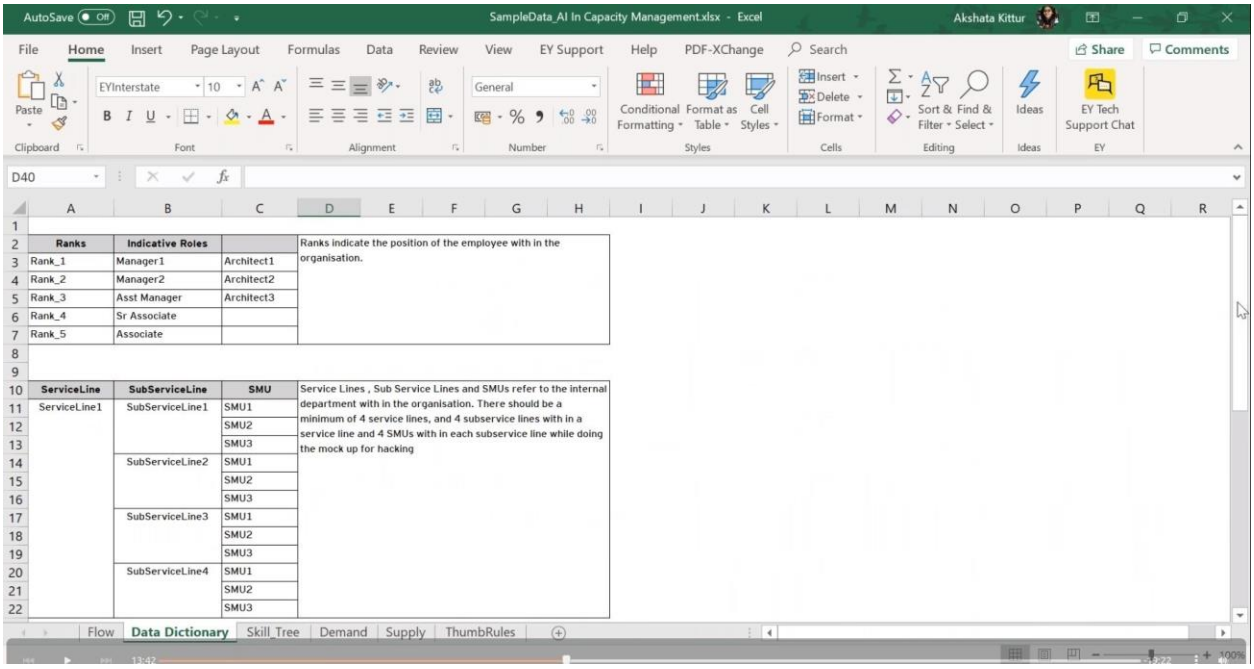
Name/ID	Primary Unit	Sub Unit 1	Sub Unit 2	Sub Unit 3	Skill	Skill Level	Years of experience	Rank	Service Line	Sub Service Line	SMU	Country	City	Bench Ageing (weeks
Employee 1	Unit 1	Technology	Emerging Trends		RPA	2	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 3	Technology	Service Delivery	Methodologies	ITIL	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Budget Management	Budget Analysis	Budget Preparation	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Business Development		Stake Holder Management	2	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 3	Technology	Management	Execution	Agile	2	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 3	Technology	Management	Methodologies	Iterative	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 3	Technology	Managemnt	Methodologies	SDLC	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 6	Business Analysis	Critical Analysis		Problem Solving	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Budget Management	Budget Analysis	Estimating	2	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Business Transformation		Programme Leadership	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Business Transformation		Programme Delivery	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 3	Technology	Management		IT Governance	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 4	Management	Business Transformation	Change Management	Change Strategy	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 7	Basic	Soft Skills		Presentation Skills	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	
Employee 1	Unit 7	Basic	Soft Skills		Effective Communication	3	20	Rank_1	ServiceLine1	SubServiceLine1	SMU1	India	Gurgaon	

Filtering it with one employee.

As explained, there could be an employee like in this supply data, which essentially means talent-pool data.

In this, you will have multiple employees. But the same employee might possess different kind of skills, such as knowledge of multiple technologies, functional skills, etc. So, our algorithm should be built in such a way that for an incoming demand, it is able to match across multiple skill sets of employees.

We will now discuss about ***data dictionary*** and its various aspects.



Ranks	Indicative Roles		
Rank_1	Manager1	Architect1	Ranks indicate the position of the employee with in the organisation.
Rank_2	Manager2	Architect2	
Rank_3	Asst Manager	Architect3	
Rank_4	Sr Associate		
Rank_5	Associate		

ServiceLine	SubServiceLine	SMU	
ServiceLine1	SubServiceLine1	SMU1	Service Lines , Sub Service Lines and SMUs refer to the internal department with in the organisation. There should be a minimum of 4 service lines, and 4 subservice lines with in a service line and 4 SMUs with in each subservice line while doing the mock up for hacking
		SMU2	
		SMU3	
	SubServiceLine2	SMU1	
		SMU2	
		SMU3	
	SubServiceLine3	SMU1	
		SMU2	
		SMU3	
	SubServiceLine4	SMU1	
		SMU2	
		SMU3	

So, essentially it denotes your rank as 1,2,3,4,5 and gives you an indicative role as a manager for 1,2 to the lowest level of employees as 5. Service Line and sub-service line essentially denotes our internal departmental structure. So, here you could have 4 service lines. For each service line, you could have another set of 4 sub-service lines. And for each sub-service line, you could have another set of SMU, so that you can mockup that data when you build that algorithm. This is one of the preferred way, to give you an indication on this.

The third thing that is quite important is the **skill tree**. For example, I've mentioned here, 'Unit 3', 'Technology', 'Management', 'Methodologies', 'Agile' followed by another skill. So, what we have is a skill tree (refer the third tab).

24	Primary Unit	Sub Unit 1	Sub Unit 2	Sub Unit 3	Skill	Skill Tree is available in the next tab - This indicates that an employee can have skills cutting across different branches (eg: Technology, Management etc.). Some skills are
25	Unit 3	Technology	Management	Methodologies	Agile	
26	Unit 6	Business Analysis	Critical Analysis		Problem Solving	

	A	B	C	D	E
1	Primary Unit	Sub Unit 1	Sub Unit 2	Sub Unit 3	Skill
2	Unit 1	Technology	Cloud Technology	Containers	Docker
3	Unit 1	Technology	Cloud Technology	Containers	Kubernetes
4	Unit 1	Technology	Cloud Technology	PaaS	Database management
5	Unit 1	Technology	Cloud Technology	PaaS	No SQL
6	Unit 1	Technology	Cloud Technology	PaaS	Application Development
7	Unit 1	Technology	Programming	Scripting	Python
8	Unit 1	Technology	Programming	Scripting	Shell script
9	Unit 1	Technology	Programming	SQL	
10	Unit 1	Technology	Programming	Scripting	Power Shell
11	Unit 1	Technology	Platform	Cloud	Azure
12	Unit 1	Technology	Platform	Windows Servers	Administration
13	Unit 1	Technology	Platform	Cloud	AWS
14	Unit 1	Technology	Platform	Unix	Administration
15	Unit 1	Technology	Web Technologies	Java	
16	Unit 1	Technology	Web Technologies	C#.net	
17	Unit 1	Technology	Web Technologies	Web development	React
18	Unit 1	Technology	Web Technologies	Web development	Angular
19	Unit 1	Technology	Data Warehousing	ETL	IBM DataStage
20	Unit 1	Technology	Data Warehousing	SAP HANA	
21	Unit 1	Technology	Data Warehousing	Business Intelligence	Power BI

Flow | Data Dictionary | Skill_Tree | Demand | Supply | ThumbRules

Here, we have come up with some set of technological skill trees on our own. You can also build your own skill tree for this particular theme. In this skill tree, we mapped some of the technical skills with functional skills and process skills against a particular unit. We followed a tree structure here, so in some cases, you will not have 'sub-unit 3' or 'sub-unit 2'. You have a mix and match your skills across multiple branches. 'Sub-unit 1', 'sub-unit 2', 'sub-unit 3' represent different branches. So, spreading across different branches you have multiple skills.

Skill tree is essentially what is seen in the supply data when we filtered with one employee. You could find that the same employee has multiple skills. So, basically, we have a skill tree against which we will map the supply data.

Next, comes the ***skill level***. Skill level is ranked from 1 to 4, with 1 being the lowest skill, and 4 being the highest skill for a particular employee.

28	**Skill Level**	Refers to the level of skill the employee possess(proficiency). This varies from 1 to 4 with 1 being lowest and 4 being highest.
29	1	
30	2	
31	3	
32	4	

Your country, and city refers to the kind of the city of requirement, or the location of the employee.

Next, comes ***years of experience***. The number of years of experience a particular employee possesses or the minimum years of experience that is required in demand data.

Next, it's ***bench aging***. That denotes the number of weeks an employee has been in the supply of talent. If that is a criteria, then it can also be taken for mapping.

Thumb rules:

1. What we intend is that our matching algorithm should be able to identify the related skills.
    - For example, if 'deep learning' or 'machine learning' is considered as a demand input, your algorithm should be intelligent enough to identify the related skills, like 'artificial intelligence'. Another example, 'agile', 'scrum masters' or 'scrum certified person' considered as input, the algorithm should have an inbuilt intelligence which will be able identify employees with the related skills
2. Next is the sample weightage given to a particular service line. You could give a weightage of location as the first priority. Then prioritize the skills. And third is experience and bench aging. You can give a rank or weightage against each of these six factors, which you mentioned.
    - You can then refine your search according to this category. This should be your weightage for finding the required set of employees, and also to find the fitment ratio, like, how against that particular demand, that employee is matching.
3. The third point covers the skill and skill level. If you assume that you have two different employees possessing the same skill where one person has a skill level of 3 and another of skill level 4, it indicates that the employee with skill level 4 should have a better fitment ratio than the skill level 3.
4. Last point focuses on the demand sheet. There exists a requestor service line, sub-service line and SMU.
    - Example: Say you have hundred results, out of which you have 5 employees belonging to the same SMU/ sub-service line and service line. Then, they should be listed first. This is followed by a set of employees who are in a different SMU but fall under the same sub-service line. They should come in next and followed by the set of employees who follow the same service line but are of different sub-service lines and SMU, it should come in the third set. So, the first five should be of SMU level match. Second five is your sub-service line level match, and then third one is your service line level match, followed by all the other search results sorted by the fitment ratio. This is done so that we're able to figure out the set of employees within the SMUs, service lines and sub-service lines first and then move to the other business units, that we can give priority to those.
5. All search results and resource matching happen across cross-service lines in the real world, as long as the skills match. We can also internally map employees from other business units as long as the skills get matched.

6. The supply data for one employee could be of multiple skills. Therefore, you should mockup or build your data in such a way that it maintains the same kind of structure to align to our business problem.
7. Fitment %: It is mandatory that some of the treatment percentage might be like 0.1% or 1.5%. In that case, we expect any and every fitment percentage more than 0% rates to get listed in the output.
8. The demand sheet contains multiple skills like 'Technical Skill 1', 'Technical Skill 2', 'Technical Skill 3', etc. You need to use an '***OR'*** condition here for dashboarding purposes (Power BI, SQL, etc.) so that you will get a maximum set of employees from your resource pool, or and then you will be able to run and match your algorithm.

Technology stack

1. Solution should ideally suit for deployment in Azure cloud platform:
    - The tools can leverage open source/ custom libraries/ Application Program Interface (API) from cloud platforms.
    - Existing Software As A Service (SaaS)/ hosted solutions (Google Cloud Platform, Amazon Web Service etc.) except Azure should not be leveraged.
2. Preferable technology stack is C#, .net and Python for Artificial Intelligence (AI)/ Machine Learning (ML).
    - You are also free to use libraries that are not licensed under Affero General Public License (AGPL).

Q&A

1. Does the skills data in supply tab essentially maps to your internal skill tree database?

Ans: The current data provided is a mockup. In real world, whatever skills defined in skill tree would be the skills in the supply. Please note, demand skills need not exactly match the skills available in the skill tree. Hackers are free to mockup their own skill tree, demand and supply data for the purpose of solution design and development

2. What is the difference between functional and technical skills?

Ans:

- Functional skill: Skills/ knowledge that one possesses for a particular domain of work – Risk Analysis, Business Analysis, Finance, Accounting etc.
- Technical skill: Technical skills refer to the technical knowledge one possesses to resolve any problem technically – programming languages, Microsoft Office, SQL etc.
- Process skill: Skill/ knowledge that one possesses to resolve/ accomplish any task – problem solving, communication skills, analytical skills, etc.

3. Can I assume that you control any entry into skill tree database?

Ans: Current skill tree is only for illustration purposes and hackers are free to develop their own skill tree, supply and demand data. Please note, the structure defined in the mockup has to be adhered in order to address the problem as expected.

4. Can search factors be added or will they be constant?

Ans: Search factors have been defined as the demand heuristics that the algorithm should follow while displaying the recommendations. Sample search factors and match criteria have been given in demand sheet and also in thumb rules sheet.

5. **Is this a classification model or recommendation model, that has been asked to build?**

Ans: You are free to interpret the problem statement as either classification/ recommendation model, however, expected outcome is to arrive at the fitment percentage and use it for classification.

6. **Any specific technology is expected for building UI?**

Ans: We are not looking for building any UI, however, basic HTML layout would be a good to have as part of the solution. Expected solution is to have an intelligent algorithm that matches demand and supply based on the search criteria or demand heuristics.

7. **Will the demand list be constant?**

Ans: No. As part of the solution demonstration, one sample demand list would work. However, we expect a scalable solution that should display results even for the changing demand lists (solution evaluation).

8. **Should the search factors be in percentage or should it be ranked as high/ medium/ low?**

Ans: Hackers are free to have their own ranking criteria but should have capability of giving precedence to one factor over the other.