Multidisciplinary Journal of TUM 2(1) 2023 35-44 DOI: https://doi.org/10.48039/mjtum.v2i1.45 **Original Article**

A Model for Mapping Graduates' Skills to Industry Roles: Machine Learning Architecture

Fullgence Mwachoo Mwakondo[1*] and Mvurya Mgala

Institute of Computing and Informatics, Technical University of Mombasa, P.O. Box 90420 – 80100, Mombasa, Kenya

*Corresponding author's Email: mwakondopoly@gmail.com

Abstract

This paper presents a machine learning architecture of a hierarchical model for mapping skills to industry roles. Currently, researchers have been approaching the problem of selecting industry roles for potential employees using flat and top-down methods. Practically, top-down approach is not reliable because it negates the natural mobility of employees in the occupational industry role hierarchy while flat approach does not take advantage of not only the easier learning property of hierarchical approach but also the local information of parent child relationship for better results. The machine learning architecture has been an attempt to address this gap using experimental research design. The mapping model consists of a collection of objects that are hierarchically arranged to progressively group industry role constructs before applying bottom-up approach to select the best. The mapping begins by first selecting the most promising sub-objects at the lower levels before passing this information to the higher levels of the hierarchy to select the most promising functional (main competence), proficiency and specialty (specific competence) objects and eventually the respective constructs. The end product is an effective machine learning architecture of a model for mapping graduates' skills to industry roles with relevant attributes to easily work with in the academia and that correctly reflects the hierarchy of industry roles. Findings reveal while SVM (67%) optimizes the model's accuracy better than naïve Bayes (57%), on the same benchmark dataset the model recorded better performance (85%) than reported performance (82%) in the benchmark model. The findings will benefit industry by getting evaluation tool for revealing information on graduate's suitability for employment which they can use for decision making when filtering candidates for interview. Besides, this will provide researchers better understanding of the gap between the academia and industry and can use this information to plan on how to bridge the gap using the mapping model. Lastly, this will attempt to reduce both low job satisfaction and long-term unemployment that is one of the causes of social and economic pain both in Kenya and around the world. However, this paper recommends testing this approach with other alternative machine learning techniques as well as other alternative industry domains.

Key words: Machine learning, Skills mapping, Hierarchical model, Bottom-up

Introduction

Machine learning is rapidly gaining popularity as a modern approach for designing models for mapping graduates' skills to industry roles yet there is very little research towards this area (Chien & Chen, 2008; Jantawan & Tsai, 2013). One of the key aspects of machine learning in multi-classification problems, that promises significant improvement to skills mapping accuracy, is the underlying machine learning architecture. The architecture determines the organization of a collection of model objects into a structure that enables efficient learning and recognition of skills patterns required by various industry roles. Usually, industry roles in most industry occupations are hierarchically structured as revealed by the four types of role organization structures namely product, geographical, functional, and matrix organizations (Malone, 2007) where the natural mobility of employees is vertically upward (NOC, 2011). This suggests that skill mapping is a bottom-up structured problem.

Currently, researchers have been approaching this problem using flat and top-down methods to select the best industry role for a potential employee (Chien & Chen, 2008). Practically, top-down approach on a bottom-up structured problem is not reliable. This is because it leads to not only multiple labels problem but also negates the natural mobility of employees in the occupational industry roles hierarchy, while flat approach does not take advantage of easier learning property of hierarchical approach (Barbedo & Lopes, 2007; Silla & Freitas, 2011). Traditionally, top-down method should be applied on a classification problem whose underlying taxonomic structure is asymmetric and transitive (Silla & Freitas, 2011). However, the transitive property of the underlying taxonomic structure exposes it to multi-labels problem when the structure is explored vertically upward using bottom-up method (Barbedo & Lopes, 2007). Besides, flat method ignores the class hierarchies and solves the classification problem by simply considering only the leaf nodes while ignoring non-leaf nodes. This may involve building a classifier to handle a large number of classes without taking care of the parent – child relationship in the class hierarchy (Wang & Casasent, 2009; Silla & Freitas, 2011).

Existing methods to skills mapping are based on flat and top-down approaches when selecting the industry role for an employee, and yet skill mapping is a bottom-up structured problem. Practically, top-down approach on a bottom-up structured problem may not be reliable, because it leads to not only multiple labels problem but also negates the natural mobility of employees in the occupational industry role hierarchy. Likewise, flat approach does not only take advantage of easier learning property of hierarchical approach but also requires a large number of classifiers. Bottom-up friendly taxonomic structure and a machine learning model with bottom-up architecture provides a potential to address both problems emanating from flat and top-down approaches. Therefore, a new machine learning architecture that tightly links all the four taxonomic structures of occupational industry roles and obeys the natural mobility of employees in the organizational hierarchy is proposed. The research hypothesis anticipates that the proposed model's architecture under appropriate machine learning technique will significantly enhance accuracy as compared to existing similar architectures.

The main focus of this paper is therefore, to design a machine learning architecture that embraces not only a taxonomic structure that represents the problem in its natural bottom-up form, but also a method that explores the structure vertically up from the bottom. The current work attempts to extend on the work of both Silla and Freitas (2011) and Barbedo and Lopes (2007) by adding, on the list of hierarchical trees for multi-class problems, a new taxonomical structure and improving on the accuracy of bottom-up method with a new machine learning architecture respectively.

Skill Mapping

The concept of industry roles is linked to the concept of occupation which is a collection of jobs, sufficiently similar in work performed and grouped under a common label known as occupational title (NOC, 2011). Some occupations are broad while others are specializations within occupational area. In addition, occupational industry roles are well defined and structured hierarchically into one of the four types of organization structures namely product, geographical, functional, and matrix organizations (Malone, 2007), and are associated with a certain skill level and type as well as occupational mobility of employees being vertically upward the structure. The four types of role organization structures are illustrated (Figure 1). As a result, computationally, skill mapping problem can be viewed as a pattern recognition problem and modeled as a machine learning (ML) task by mapping skills to predefined roles in the hierarchical structure and learn a model to classify graduates' skills from bottom to top. For this solution to work effectively, a suitable ML architecture must be designed and trained to classify industry roles according to predefined set of industry roles.

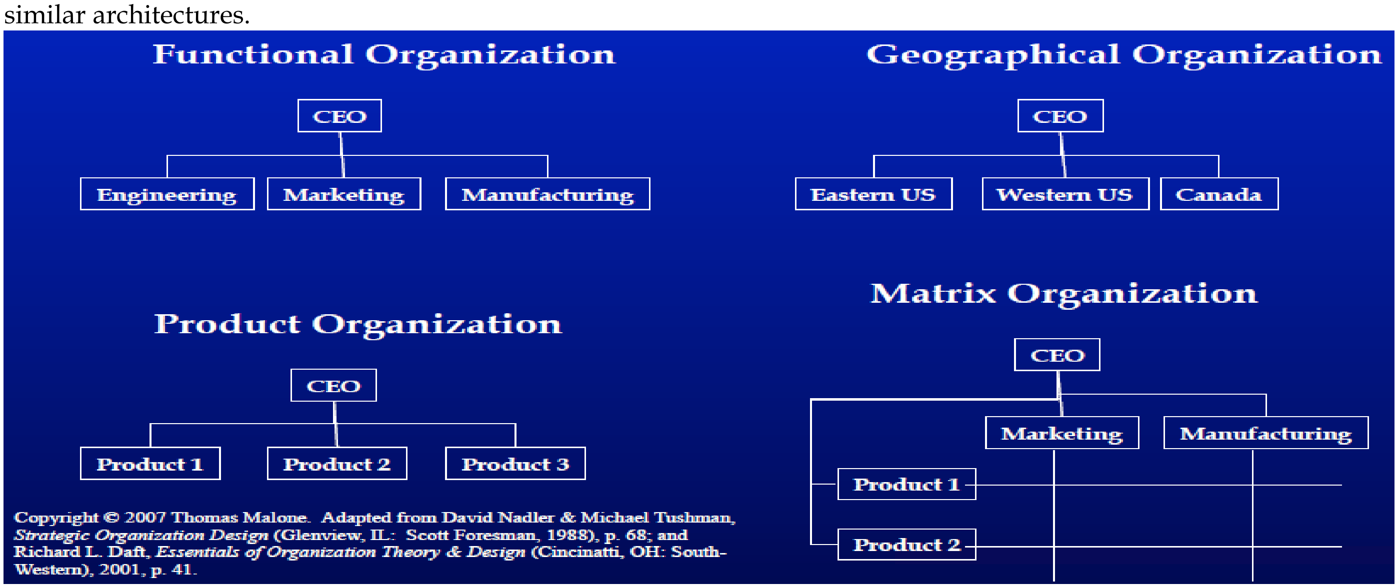

Figure 1. Role organization structures

Top-down *versus* Bottom-up Approaches

Top-down Approach

In top-down approach, a problem is split repeatedly into smaller units and each unit is further split over and over again until the resulting smaller problem unit is manageably solved. The main idea is to solve the problem progressively from generality (complexity) to specificity (simplicity) where the underlying problem is described hierarchically using a tree structure that is asymmetric and transitive (Silla & Freitas, 2011). In the classification problem, top-down method is used to first predict the most generic class (generic level) then it relies on the predicted class to select the next level class where the only valid candidate classes are children of the previous level predicted class, and this is repeated in each level until the most specific class is predicted. Two common types of taxonomic structures for machine learning that support top-down approach as tree and di-acyclic graph (DAG) according to Silla and Freitas (2011) are presented (Figure 2).

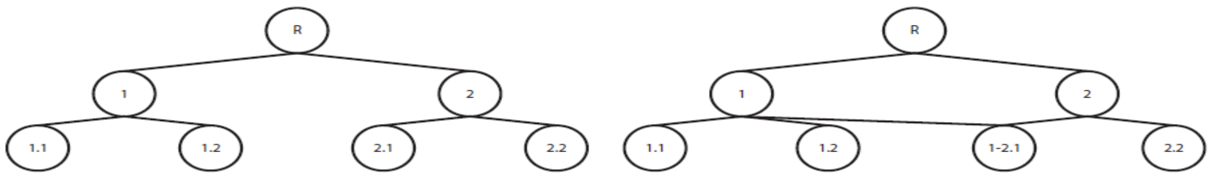

Figure 2. Tree structure (left-side diagram) and DAG structure (right-side diagram)

According to Silla and Freitas (2011), most hierarchical classification problems are based on tree or DAG structures whose "IS-A" relationship is asymmetric, anti-reflexive, transitive, and has the following properties:

1) The only one greatest element R is the root of the tree.
2) For every class c_i; $c_j \epsilon C$; *if* c_i *is related to* c_j *then* c_j *is not related to* c_i.
3) For every class $c_i \epsilon C$; c_i *is not related to* c_i.
4) For every class c_i; c_j; $c_k \epsilon C$; c_i *is related to* c_j *and* c_j *is related to* c_k *imply* c_i *is related to* c_k.

Coincidentally, the above structures have been used for both top-down and bottom-up approaches. However, for bottom-up, there are challenges with consistency of class membership in the hierarchy and, therefore, are only suitable for top-down approach, where the classification is approached from general to specific. In addition, none of the above machine learning structures can represent all the four underlying organization structures of industry roles. For example, tree structure is only appropriate for functional, product, and geographic types while DAG is appropriate only for matrix type. Clearly, a machine learning structure that tightly links all the four taxonomic structures of occupational industry roles and that obeys the natural mobility of employees in the organizational hierarchy is needed.

Bottom-up Approach

In bottom-up approach, the problem solution is derived in the reverse order of top-down approach (Barbedo & Lopes, 2007). The main idea is to analyze large number of specific items (simple) so as to find relationships and patterns that can help to generalize into a meaningful item (complex). The aim is to solve the problem progressively and incrementally from the most specific (simple) and basic aspects to the most complex and generic solution. This approach involves both lower level local processing and higher-level global processing, where lower level specific/basic items are analyzed to provide information that helps to generalize into meaningful and complex higher-level items (Maloof, 1999; Amir, 2014). However, the underlying structure of some problems may not be fit for top-down approach but bottom-up approach. Besides, applying a bottom-up method on the traditional taxonomic tree structures as defined by Silla and Freitas (2011) leads to either class inconsistency or multiple label classification problems as revealed by Barbedo and Lopes (2007). As a result, the current paper proposes not only a machine learning architecture for a skill mapping model but also a taxonomic structure that is bottom-up friendly and that tightly links all the four taxonomic structures of organizing occupational industry roles.

Related Work

World Economic Forum report (2018) on the Future of Jobs points at skills gap among workers and leaders in the organization as likely to hamper technological adoption as well as business growth. The skills gap has been as a result of technological breakthroughs that have rapidly shifted the way work tasks are performed by humans into a new way they can be performed by

Published, January 2023

machines and algorithms. This transformation of jobs has resulted into large scale decline of some roles as well as large scale growth of new roles associated with adoption of new technology. Three strategies to address the skills gap have been outlined as: 1) hire wholly new permanent staff with skills relevant to new technology, 2) automate the concerned work tasks completely, or 3) retrain the existing employees. However, augmentation strategy has been favored where some tasks are automated to complement and enhance strength of human workforce and that empowers workers to extend to their full potential. To achieve this, it requires hiring workers with appropriate skills and proficiency in the new technology so as to enable them to thrive in the work place of the future as well as ability for lifelong retraining.

Large number of graduates hold jobs that do not make best use of their skills (70% in sub-Saharan Africa; 35% in Europe, ILO, 2015). Revelation in the literature indicates fewer studies towards skills mapping using not only machine learning techniques (Chien & Chen, 2008; Jantawan & Tsai, 2013) but also bottom-up approach. Zaharim et al. (2010), applied requirements of professional bodies and accrediting bodies to construct a skills mapping framework for Malaysian Engineering graduates. Chien and Chen (2008) built a skill mapping model using data mining techniques for prediction of employee retention of new job applicants. They all used flat approach. Jantawan and Tsai (2001) presented a skill mapping model for predicting graduate employment twelve months after graduation based on flat approach. Many of these studies approach skills mapping to industry roles using flat or top-down method yet natural mobility of employees in the industry is bottom-up. Different trends of underlying classification structure used for machine learning in skills mapping are presented (Table 1).

Table 1. Trends in machine learning structure for skill mapping

Author/work	**Year**	**Method**	**Type of Attributes**	**Classification**
Chien & Chen	2008	Classification	Demographic profile	Flat
Jantawan & Tsai	2013	Classification	Demographic profile	Flat
Korte et al.	2013	Classification	Qualifications	Flat
Srikant & Aggarwal	2014	Regression	Programming practices	Flat
Shashidhar et al.	2015	Classification	English, Logical,	Flat

Proposed Machine Learning Architecture

Our proposed machine learning architecture consists of three sections: a) taxonomic structure, b) architecture of the mapping model, c) architecture of the basic model's classifier objects. The proposed underlying taxonomic structure together with the bottom-up method allows the model objects to train the 'children to recognize their parents' and not *vice-versa* as is the case of top-down approach (Wang & Casasent, 2009; Silla & Freitas, 2011). The children are trained to recognize their parents at different levels of hierarchy. The basic idea is to explore the underlying taxonomic structure from the bottom to top as naturally as required by some problems, such as skill mapping to industry roles. To achieve desired results, the model objects' design adopts the 'sibling' policy during the training where siblings of same parents are trained against siblings of other parents using one-against-all binary model objects.

Proposed Bottom-up Friendly Taxonomic Structure

Generally, in supervised machine learning the output of each model object should be defined over a taxonomic structure of classes (Silla & Freitas, 2011). For skill mapping, the classes are industry roles and each role is characterized by three dimensions namely main competence, specific competence, and proficiency (CWA16458, 2012). The bottom-up friendly taxonomic structure (BFTS) that represents the three dimensions graphically as the hypothetical structural organization of role classes as per the structured classification problem and classification assumptions in this method is proposed (Figure 3). This figure illustrates hierarchical structure with two branches (may be more), each branch with three levels, a total of twelve leaf node classes (C1.5, C1.6, C1.1.3, C1.2.4, C1.2.1, C1.2.2, C2.5, C2.6, C2.1.3, C2.1.4, C2.2.1, and C2.2.2), and a total of six parent nodes (1, 1.1, 1.2, 2, 2.1, and 2.2), and root node (R). Leaf nodes represent specialized individual roles while the upward arrow indicates the direction of employees' occupational mobility. However, although the proposed taxonomic structures "IS-A" relationship is asymmetric and anti-reflexive as in Sillas and Freitas (2011) definition of "IS-A" relationship, it departs away from this definition by being anti-transitive with the following properties:

1) The only one greatest element R is the root of the tree.
2) For every class c_i; $c_j \in C$; *if* c_i *is related to* c_j *then* c_j *is not related to* c_i.
3) For every class $c_i \in C$; c_i *is not related to* c_i.
4) For every class c_i; c_j ; $c_k \in C$; c_i *is related to* c_j *and* c_j *is related to* c_k *does not imply* c_i *is related to* c_k.

As per the assumptions of the current problem statement, each branch represents sub-occupation, each non-leaf node represents main competence, and each leaf node represents specific competence, while each level represents proficiency. However, while each main competence belongs to a certain proficiency level, each proficiency level in each branch is associated with only one main competence. Thus, relationship between main competences is one of peer to peer. As a result, these concepts have been applied in subsequent discussion of the proposed machine learning architecture. The main difference between the proposed taxonomic structure and the traditional tree structure is eminent at the levels/non-leaf nodes where the former adopts peer-to-peer and the later adopts parent-child relationships. While in the traditional structure lower level parents are decompositions of higher-level parents, this is not the case in the proposed structure as each level is a category that indicates superiority of skill proficiency.

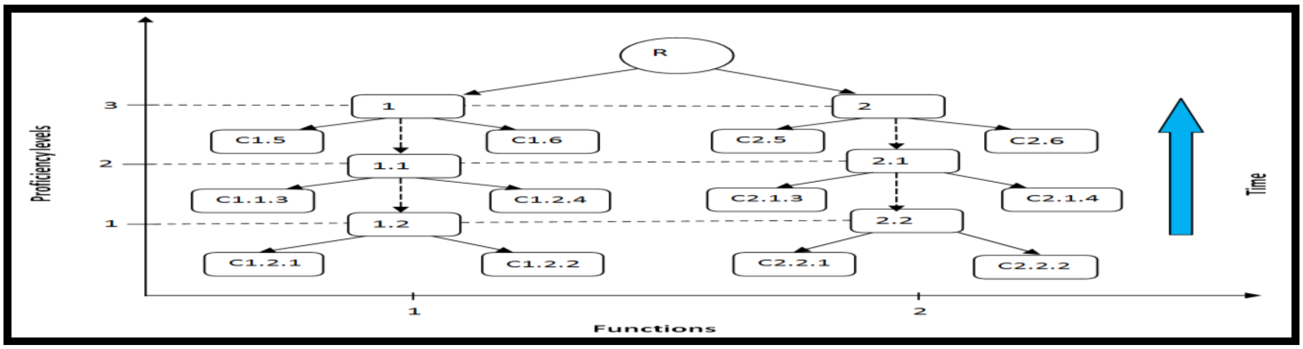

Figure 3. Bottom-up friendly taxonomic structure

Proposed Machine Learning Architecture for Skill Mapping Model

Figure 4 illustrates a machine learning architecture of a model for exploring the proposed taxonomic structure in Figure 3. The mapping model consists of a number of objects that are hierarchically arranged to progressively group industry role constructs before selecting the best. At each level, different kind of objects are triggered to generate specific type of information about industry role construct that is jointly used at the higher level for further processing and this continues up to the highest level where the most promising role class is predicted. The model objects at lower level gather local information about the potential sub-occupation which they then pass to higher model objects to collect further local information about the potential proficiency and eventually the potential specific competence.

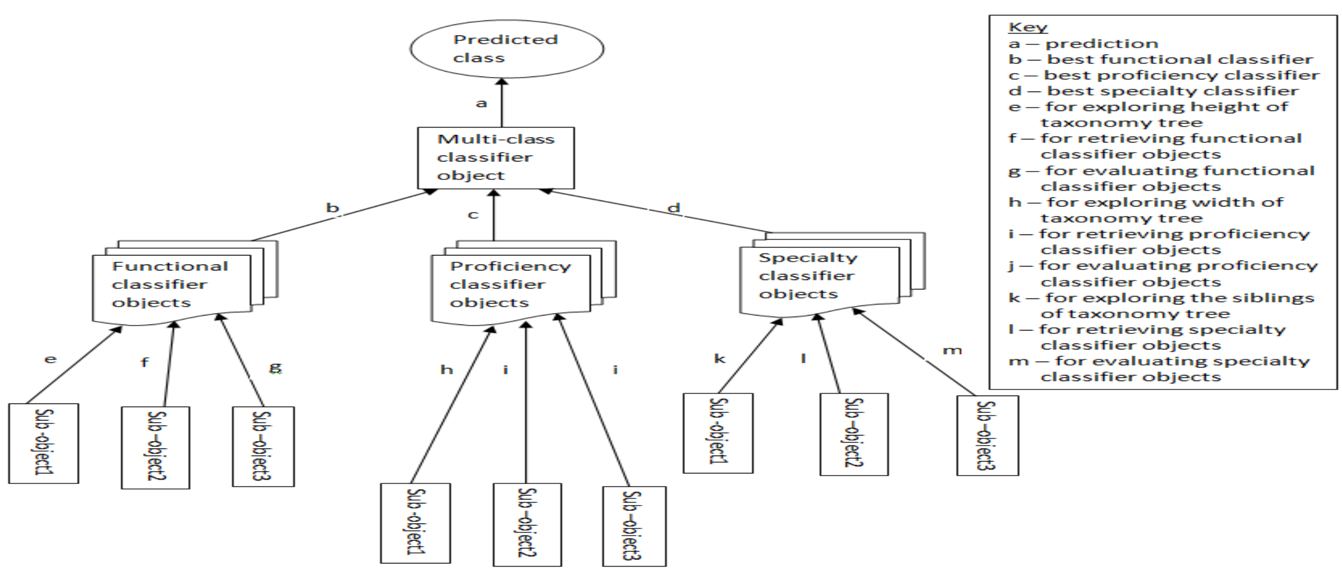

Figure 4. Machine learning architecture of skill mapping model

Published, January 2023

Basic Architecture of Model's Classifier Objects

Machine learning is one of the commonly representatives of bottom-up analysis where various types of data are analyzed to reveal relationships and patterns (Wirsch, 2014). As a result, the underlying structure of each machine learning object is based on bottom-up method. The conceptual framework in our previous work Mwakondo et al. (2016a) provided the original basic machine learning architecture of the model's classifier objects.

Research Methodology

Experimental design seemed to be the most appealing research method to demonstrate the practicability of this approach. As such, a case of a bottom-up hierarchically structured multiclass problem, such as skill mapping to industry roles was preferred. Therefore, a domain of industry occupations was adopted as a suitable ground, to demonstrate our approach, in which several jobs sufficiently similar in work performed are grouped under a common occupational title (NOC, 2011). Software engineering was selected as a typical industry occupational domain where there are several similar jobs grouped together as software engineers (Suraka, 2005). The compositional structure of software engineers' industry roles is hierarchical and recognition of each of the role requires not only both local and global processing but also bottom-up exploration of the structure. Our previous work (Mwakondo et al., 2016b) revealed the compositional structure of software engineers' industry roles which conforms to the currently proposed taxonomic structure and provided the dataset for experiments.

Initially, a prototype was built of a skill mapping model based on the proposed bottom-up machine learning architecture to demonstrate how skill profile from employees can be used to derive a single-label prediction model to map graduates' skills to industry roles. In this case, two machine learning techniques (naïve Bayes and Support Vector Machines) and three datasets for employees' skills profile were involved in the experimental investigation. Experiments to validate the model were designed using repeated 5-fold cross validation technique before its performance was evaluated using test data. A framework adopted for repeated 5-fold cross-validation technique is outlined (Figure 5). Initially the model's performance was validated using SE datasets, both field and benchmark, then evaluated using three test datasets namely, SE field, SE benchmark, and Academic Librarians (AL) datasets. Model's validation involved selecting the best features, optimal parameter values, and the best induction algorithm for the model. Performance results reported on carefully selected benchmarks on bottom-up multi-classification method was adopted for results validation. Currently, bottom-up method has not been applied in skill mapping to industry roles. However, in other domains, such as music genre classification, bottom-up method has been used successfully. And especially, in the work of Barbedo and Lopes (2007) where average performance of 61% in the leaf nodes was reported, provided a benchmark for comparison.

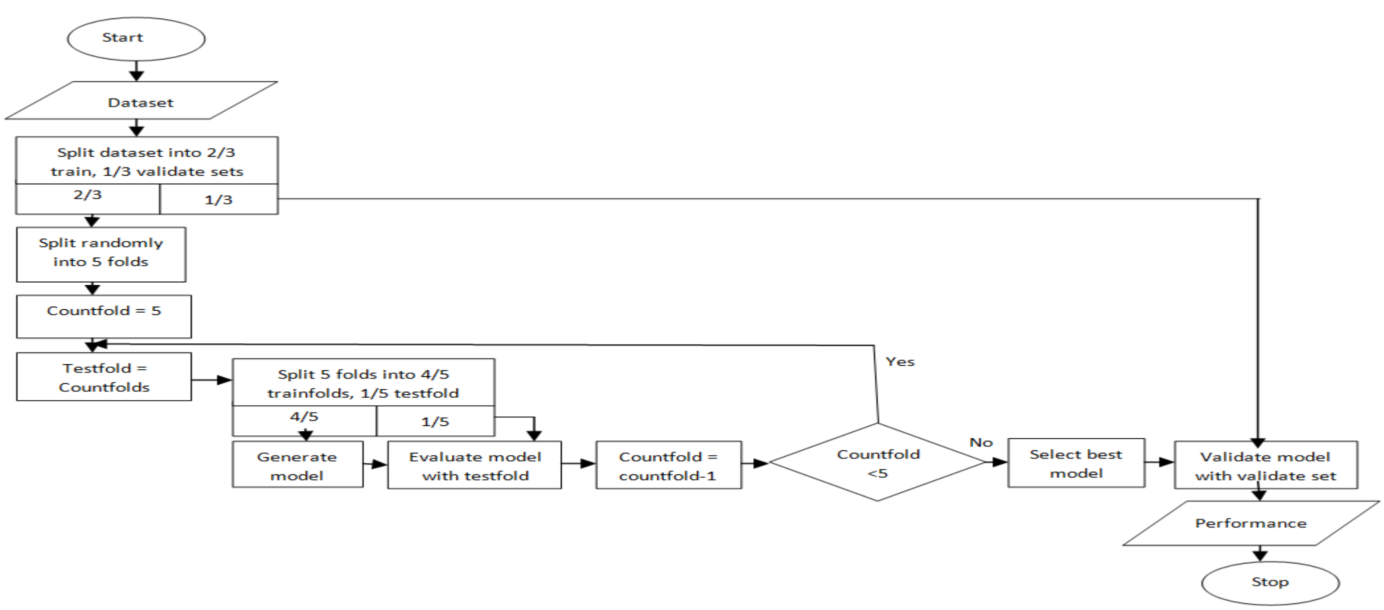

Figure 5. Experiment execution model adapted from Clare & King (2003)

Results and Discussion

The demographic characteristics of the experimental datasets used in the investigation are presented (Table 2). Dataset2 was used as benchmark dataset where Shashidhar et al. (2015) on same dataset using a related model reported performance accuracy of 82%. Dataset1 and 3 were used as multiple case studies for different industry domains to validate model's results for generalizability. The model was generated using two induction algorithms, hence two versions of the model. The two models were experimented under similar conditions and results compared. This involved fitting and testing both models with similar training and validate sets respectively through 10 iterations of 5-fold cross-validation.

Table 2. Demographic characteristics of experimental datasets

Dataset	Attributes	Instances	Classes	Levels
1. Dataset1 (SE field)	18	113	12	3
2. Dataset2 (SE benchmark)	18	279	12	3
3. Dataset3 (AL field)	14	50	7	3

Results of this experiment indicated that there was a difference in mean performance between SVM and naïve Bayes models (56.79, 52.54 in dataset1 and 78.77, 63.93 in dataset2, respectively; Table 3). This suggests that SVM model was better than naïve Bayes. Further investigation was conducted to test whether the differences (4.25 and 14.84) were real and significant. This test was conducted using paired sample T-test procedure. This test was conducted to test the hypothesis that model performance difference was not significant. For this type of test to be valid, conditions for tests were checked (homogeneity and normality of data). The results indicate the difference was real and significant ($p > 0.05$ in both cases).

Table 3. Model's performance validation

Test fold		***5-Fold cross validation accuracy tests (%)***			
		Naïve Bayes	*SVM*	*Naïve Bayes*	*SVM*
Fold_1	Mean	60.81	73.30	49.51	55.86
	N	10	10	10	10
	Std. Deviation	3.38	3.65	7.54	9.59
Fold_2	Mean	63.00	77.78	48.89	59.41
	N	10	10	10	10
	Std. Deviation	3.70	4.21	11.37	4.99
Fold_3	Mean	66.69	80.18	56.22	58.09
	N	10	10	10	10
	Std. Deviation	5.92	6.04	7.22	10.65
Fold_4	Mean	63.35	81.90	51.22	53.11
	N	10	10	10	10
	Std. Deviation	5.49	2.63	10.54	8.46
Fold_5	Mean	65.79	80.70	56.84	57.48
	N	10	10	10	10
	Std. Deviation	6.18	5.81	13.97	12.70
Total	**Mean**	63.93	78.77	52.54	56.79
	N	50	50	50	50
	Std. Deviation	5.30	5.42	10.56	9.48

To confirm the difference was not due to any other factor but only machine learning construct difference, ANOVA test was conducted to rule out the effect of fold to fold variations. Results of ANOVA analysis for both kinds of model constructs indicate the fold variances were equal and, in fact, means of the fold scores were not different and therefore the seemingly difference between the two models in Table 3 was not due to effect of fold to fold variations ($p < 0.05$ in all cases). This was enough reason to select SVM model as the best classifier. Finally, we needed to test the quality of our model using appropriate quality measures. This was after realization that accuracy alone sometimes could be misleading as sometimes a model with relatively high accuracy was likely to predict the 'not so important class labels' fairly accurately while making all sorts of mistakes on classes that were actually critical. As a result, other performance measures such as precision, recall and F1 scores were incorporated. The aim was to study the ability of the model to find all the positive instances correctly (recall) and ability not to label negative instances as positive (precision) or weighted average score of the two (F1). Table 4 illustrates results of the model performance along four quality metrics and across three datasets, while Table 5 presents performance results along hierarchical levels across the three datasets. In each case, the model reported equal performance in both accuracy and recall. However, its ability not to label negative classes as positive was not as good as its ability to find all positive classes correctly which was equally good (precision = 66%, recall = 69%). On average, model performance seemed to improve upward the hierarchy levels consistent with other models in literature (Clare & King, 2003; Barbedo & Lopes, 2006). Model's performance seemed to be very high in the benchmark dataset as a result of having more instances whose classes had very high accuracies (class 10 & 11) and fewer instances whose classes had very low accuracies (class 7 & 8). This was not the case with other two datasets where distribution differences of classes with very high and very low accuracies were not high. In the Benchmark dataset where performance was 85%, high accuracy (100%) class (class11) had the highest number of instances (size = 11) while low accuracy (5%) class (class7) had the lowest number of instances (size = 2). In Research dataset where performance was 59%, high accuracy (93.4%) class (class7) had moderate number of instances (size = 3) while low accuracy (5%) class (class3) had moderate number of instances (size=1). In Validation dataset where performance was 65%, high accuracy (100%) class (class1&5) had moderate number of instances (size=2) while low accuracy (5%) class (class2&7) had moderate number of instances (size = 2). Model performance in both Research and Validation datasets seemed to be fairly good (59% and 65%, respectively). These results indicate the best generalization performance as an average performance calculated across the three datasets. In this case, along hierarchical levels the best average performance accuracy of the model was 67% while general average performance was 69%. Therefore, we can confidently claim that the best performance of our model was 67%.

Table 4. Comparison of performance across three datasets

Performance Metric	SE field dataset (Research)	SE lit. dataset (Benchmark)	AL field Dataset (Validation)	Mean
accuracy	0.59	0.85	0.65	**0.69**
precision	0.62	0.83	0.54	**0.66**
recall	0.59	0.85	0.65	**0.69**
F1_score	0.57	0.83	0.56	**0.65**

Table 5. Comparison of performance along hierarchical levels across datasets

	Research Dataset		**Benchmark Dataset**		**Validation Dataset**		
level	**classes**	**average**	**classes**	**average**	**classes**	**average**	**Mean**
1	7,8	0.79	1,2,7,8	0.53	4,5	0.98	**0.77**
2	3,4,9,10	0.41	3,4,9,10	0.95	3,6	0.73	**0.69**
3	5,6,11,12	0.43	5,6,11,12	0.82	1,2,7	0.37	**0.54**
Mean		**0.54**		**0.77**		**0.69**	**0.67**

Conclusion

The research hypothesized that the proposed model's architecture under appropriate machine learning technique would significantly enhance accuracy as compared to existing similar architectures. This was approached using an experimental design where the findings revealed indeed the proposed model's architecture achieved better accuracy level under SVM (67%) than even related models reported in other problem domains: 53.3% in protein classification using top down method (Clare & Kings, 2003) and 61% in music genre classification using bottom-up method (Barbedo & Lopes, 2006). On the same dataset our model recorded performance accuracy of 85% better than 82% reported by Shashidhar et al. (2015). The implication and significance of the quality of the model depends on the trade-off cost between false positives and false negatives, which is application field dependent (Maloof, 1999). In skills mapping context, the cost of false positives is much higher to the employer than the cost of false negatives to the employee. This is because of a case where unsuitable employee may be appointed and result in poor performance and low productivity in the job leading to a loss not only to an employer losing profit opportunity but also an employee risking dismissal.

This research finding is a great step forward not only in skills mapping but also in other application fields where the underlying problem structure is bottom-up, such as computer intrusion detection. The demand for such kind of machine learning architecture arises when the problem needs to be explored from low level local processing to high level global processing. In such problems, as noted by the findings we stand to achieve significant improvement in the results accuracy.

Acknowledgement

The authors greatly appreciate anonymous reviewers for the successful publication of this manuscript. We also give our gratitude to the National Council for Science, Technology and Innovation (NACOSTI) which was very instrumental in supporting the research funding of this study as part of PhD research for FMM.

References

Barbedo J. G. A., & Lopes A. (2007). Automatic Genre Classification of Musical Signals. *Journal on Advances in Signal Processing:* 64960 doi:10.1155/2007/64960

Chien C., Chen L. (2008). Data mining to improve personnel selection and enhance human capital: A case study in high-technology industry. *Expert Systems with Applications,* 34 (2008) 280–290

Clare, A. & King, R.D. (2003). Predicting gene function in saccharomyces cerevisiae. *Bioinformatics* 19, ii4-ii49, suppl. 2.

CWA16458 (2012). European ICT Professional Profiles. *CEN Workshop Agreement May 2012*. 81pp

International Labour Office. (2015). Global Employment Trends for Youth 2015: *Scaling up Investments in Decent Jobs for Youth (Geneva).* Report 2015: 98pp

Jantawan, B. & Tsai, C. (2013). Application of Data Mining to Build Classification Model for Predicting Graduate Employment. *arXiv preprint arXiv:1312.7123.*

Korte, W.B, Husing, T., Hendriks, L. & Dirkx, J. (2013). Towards a European Quality Label for ICT Industry Training and Certification. Final Report 2013.

Mwakondo, F.M., Muchemi, L., & Omwenga, E.I. (2016). Proposed Model for Predictive Mapping of Graduate's Skills to Industry Roles Using Machine Learning Techniques. *The International Journal of Engineering and Science* 5(4), 15-24.

NOC. (2011). Human Resource and Skills Development Canada. *National Occupational Classification 2011 Report. 121pp.*

Silla, C. N & Freitas, A. A. (2011). A Survey of Hierarchical Classification across Different Application Domains. *Data Mining and Knowledge Discovery ·January,* 22 (1): 31-72

Srikant, S. & Aggarwal, V. (2014). A System to Grade Computer Programming Skills using Machine Learning. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining:* 1887–1896.

Surakka, S. (2005). Trend Analysis of Job Advertisements: What Technical Skills Do Software Developers

Need? *Informatics in Education,* 4(1), 101-122.

Wang, Y. F & Casasent, D. (2009). A Support Vector Hierarchical Method for Multi-class Classification and Rejection. *In 2009 International Joint Conference on Neural Networks*: 3281-3288.

Wirsch, A. (2014). Analysis of Top-down Bottom-up Data Analysis Framework and Software Architecture Design. *web.mit.edu/smadnick/www/wp/2014-08.pdf*

World Economic Forum. (2018). The Future of Jobs Report 2018: Centre for the New Economy and Society, Insight Report: 147pp

Zaharim, A., Omar, M.Z., Yusoff, Y.M., Muhamad, N., Mohamed, A., and Mustapha, R. (2010). Practical framework of employability skills for engineering graduate in Malaysia," *In: IEEE EDUCON Education Engineering 2010: The Future of Global Learning Engineering Education:* 921–927.