DOI: 10.33965/celda2019_201911L026
16th International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2019)

DIVERSITY AS AN ADVANTAGE: AN ANALYSIS OF CAREER COMPETENCIES FOR IT STUDENTS

Patricia Brockmann[1], Heidi Schuhbauer[1] and Annika Hinze[2]
[1]*Technische Hochschule Nuernberg Georg Simon Ohm, Postfach 120320, D-90121 Nuremberg, Germany*
[2]*University of Waikato, Private Bag 3105, Hamilton 3240, New Zealand*

ABSTRACT

Due to increasing digitization in all aspects of life, the demand for qualified software development professionals continues to increase. Students from underrepresented groups, such as first generation students from non-academic families, minorities, single parents and women represent an underutilized pool of untapped potential talent. The question arises as to which unique perspectives computer science graduates from underrepresented groups can bring to software development companies. In addition to programming skills, non-technical competencies, such as foreign language abilities, intercultural communication, creativity, conflict management, team-building and organizational skills are vital for success in diverse, international project teams. A large job market database for new graduates, developed for a consortium of universities in Bavaria, Germany, is analyzed using machine learning tools. Career competencies desired by recruiting companies are compared to potential advantages offered by computer science graduates from underrepresented groups.

KEYWORDS

Education, Diversity, Competencies, Computer Science, Machine Learning

1. INTRODUCTION

The rise of digitization rapidly permeates through all aspects of modern life. As a result, the demand for qualified software development professionals continues to increase. This growing demand, accompanied by a simultaneous shortage of available software developers, widens the skills gap between job openings and qualified applicants. Traditionally, a number of groups have remained underrepresented among computer science students: first generation students, people who come from a migration background, single parents and women. People from these hitherto underrepresented groups offer an untapped source of potential talent.

In Germany, the proportion of people who come from a migration background (23%) is significantly higher than the proportion of migrants at universities (11%) (BAM 2011). At the Technical University of Applied Sciences in Nuremberg, 65% of the students are from non-academic families, approximately 5% have children, 10% are international students and 5% do not have a high school diploma (THN 2018). The number of female students in the Computer Science Department has actually been declining. In the Winter Semester of 2018/2019, only 17% of newly enrolled students were female (IN THN 2018).

Graduates from underrepresented groups can bring unique advantages to software development teams. They can help to increase the diversity of perspectives examined. Ilumoka (Ilumoka2012) discusses the importance of diversity in engineering teams. Especially during the requirements engineering phase, non-technical skills, such as intercultural communication and foreign language abilities, can be of exceptional value for multi-national teams or for stakeholders in foreign countries. During the software development and testing phases, cooperation, team-building and conflict management skills can prove vital for the success of a software project.

This work considers two research questions:

1. Which career competencies do students need to master for future careers as IT professionals?
2. Can machine learning methods help to gain information about the job market to help computer science students plan their future careers?

Section 2 Related Work discusses literature related to career competencies which are important in IT. Section 3 Methodology describes the machine learning methods used in this investigation. Section 4 Results presents the findings of the analysis. Section 5 Conclusions discusses the implications of the results, their relevance for IT graduates from diverse backgrounds and describes plans for further research.

2. RELATED WORK

This section discusses work related to the career goals and motivation of computer science students as well as the competencies students need to learn for their future careers as software developers. International, cultural and gender aspects are presented.

Liebenberg and Pietersee (Liebenberg 2016) investigated the career goals of software development students and professionals in South Africa. They found that both students and professionals valued stability and work/life balance most highly. Professionals additionally expressed the value of creativity. They assert that knowing people's motivation can help to improve recruitment and retention of software developers.

The intercultural competencies necessary to work in global software development teams have been investigated by a number of authors. Beecham, et al., (Beecham 2017) conducted a wide-scale literature review of distributed global software engineering courses. They identified a number of difficulties inherent to working in international software teams, which students need to learn to address: distance, teamwork, soft issues, stakeholders from industry, infrastructure and distributed software development processes. They categorized various types of distances, such as physical (geographic), time zones, cultural, language and institutional distances. Other authors, such as Hoda et al., (Hoda 2016) concentrated on the socio-cultural capabilities which students need to learn to work effectively in global software development teams. They pointed out the importance of overcoming language barriers, different perspectives regarding time, attitudes towards achievement, differences in autonomy and work habits as well as assumptions about national culture. They underline the importance of cross-cultural training. One example of the importance of cultural sensitivity in requirements engineering was reported by Hinze, et al. (Hinze 2018). To develop a medical app aimed at improving the health of migrant communities, sensitive personal data needed to be collected. With such a multi-cultural stakeholders, they stressed the importance of establishing personal relationships in order to create a trusting environment. Ideally, they recommend that one member of the research team should come from the cultural community studied, in order to help build bridges between the two worlds.

A number of authors have analyzed the effect of gender on computational thinking in schools and in the work place. Budinska and Mayerova (Budinska 2017) investigated the relationship between computer science concepts and computational thinking, in this case graph tasks. They found that boys were comparatively better at tasks with simple, relatively abstract representation and a larger amount of text, with the goal defined to identify a problem. They found that girls were better at tasks with less text, but with a relatively more complicated representation of structure, with a focus on simple operations on graphs. They concluded that because boys and girls have different methods of acquiring mechanical and abstract thinking, they each need different types of assignments to increase their motivation. Cheryan, et al. (Cheryan 2011) examined whether role models have an effect on self-confidence. They found that women who interacted with non-stereotypical role models believed they would be more successful in computer science than those who interacted with stereotypical role models. Faulkner (Faulkner 2009) discusses the subtle dynamics which can contribute to a feeling of 'belonging' in work relationships. She discusses the importance of informal conversation topics among colleagues, which can make women and other underrepresented groups feel like outsiders. Branz, et al. (Branz 2019) used Sentiment Analysis to evaluate how male and female team members interact on software engineering projects. They used statistical and machine learning methods to analyze a large data set from an incident management system to investigate the emotional content of project communication. They found that the types and intensities of sentiments expressed differed considerably between male and female developers.

The literature discussed here illustrate some of the challenges which computer science students will face upon graduation. The question arises as to whether students from underrepresented groups can leverage their backgrounds to make unique contributions to increase the diversity of ideas contributed to software development teams.

3. METHODOLOGY

In order to test the two questions proposed in Section 1 Introduction, an initial experiment was conducted using machine learning. To approximate the professional competencies required for future IT careers, a large database of job openings for new graduates was analyzed. The Job-Boerse (Jobboerse 2004) was developed in 2004 by the Computer Science Department at the Technical University of Nuremberg Georg Simon Ohm in Germany. Currently, 15 universities are participating in the project. For this experiment, a total of 30,792 job ads spanning over 3 years (2016 - 2018) were analyzed.

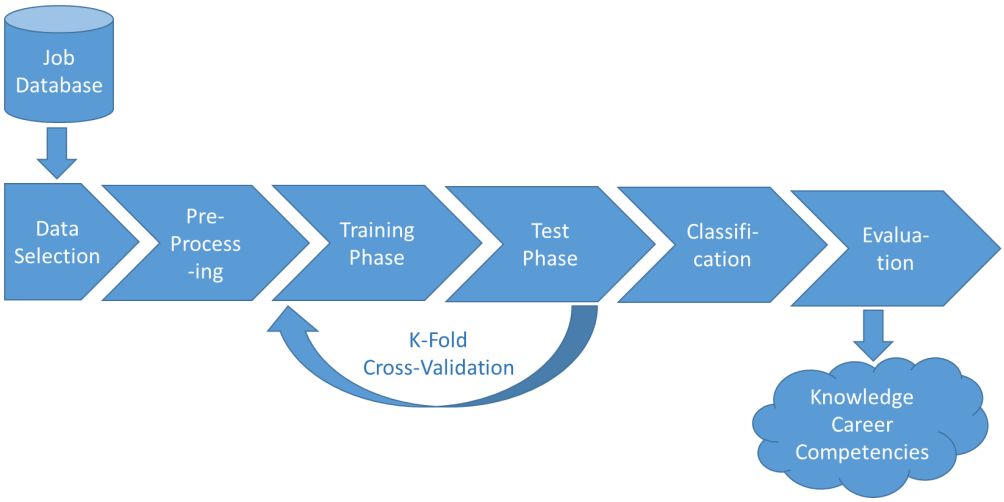

Figure 1. Machine Learning Process

First, data was extracted from the job database. Unlike highly structured database entries, individual job offers are stored as unstructured text. The first challenge was to extract the information relevant for this analysis, such as the job title and necessary qualifications, from free-form text. Next, the extracted data was cleaned in a pre-processing phase. Text errors in the database were corrected using a pre-processor written in Python, based on the FTFY library (Speer 2019). The analysis was performed using a machine learning system named Weka. The Weka system is a research project of the Machine Learning group at the University of Waikato in Hamilton, New Zealand (Frank 2016).

Three competing algorithms were tested to classify job offers:

1. Naive Bayes Classifier (NBC)
2. K-Nearest Neighbors (KNN)
3. Support Vector Machine (SVM)

A Bayes statistical classifier algorithm was used as a baseline to test whether machine learning algorithms are actually better than classical statistical methods. Bayes classifiers are based on the Bayesian statistical theorem. Data points are assigned to a given class using a set of features, called a feature vector. A naive Bayes classifier (NBC) simplifies this process by making the 'naive' assumption that all feature variables are independent of class (Rish 2001). The naive Bayes algorithm has proved effective in text classification in a number of applications (Chen 2009).

The K-Nearest Neighbors algorithm (K-NN) is an instance-based learning algorithm which is based on similarity measures between data points. K-NN is a supervised learning method which uses data from the past, with known output values, to predict an unknown output value for new data (Korde 2012). It has also been called a non-parametric, 'lazy' machine learning algorithm, because the algorithm makes no assumptions about the form of the problem. Any new generalization beyond the initial training data is first performed when each new query is encountered (Sebastiani 2002).

A Support Vector Machine (SVM) is a supervised machine learning algorithm which classifies data into groups by constructing hyperplanes. In a two dimensional space, this hyperplane would be represented by a line dividing the data points into two groups. Non-linear classification can be performed by mapping inputs

to higher dimensional hyperplanes (Tong 2001). Support vector machines have been shown to be highly effective to classify text (Basu 2003).

During the initial training phase, a randomly selected subset of the job ads was used to train each of the three algorithms to classify job descriptions into groups. These groups were defined as specific job titles, such as 'developer', 'consultant', 'engineer', 'analyst' and 'project manager'. This was followed by a test phase, during which a different subset job ads, those not used during the training phase, were tested. This test data set of new, unseen job ads was input into the trained algorithms to see whether the job titles in the new job ads could be correctly recognized by each of the three algorithms.

To estimate the effectiveness of the competing algorithms, a K-fold Cross Validation Test was performed. Each of these training and test phases were repeated for 10 cycles for each of the machine learning algorithms tested. During each test cycle, a different subset of the jobs data was used for the training phase and the remaining data for the test phase (Wong2015).

Finally, the career competencies were grouped into clusters using the K-means algorithm. The K-means algorithm partitions data objects into a specified number of groups, 'K'. Each data object is assigned to the cluster with the nearest mean value (Jain 2010). The K-means algorithm has been shown to be effective to group large text document data sets (Huang 2008). The premise here is that the competencies associated with job titles are those desired by companies recruiting new graduates.

4. RESULTS

As described in Section 3 Methodology, records were first extracted from the job database, then cleaned with a pre-processing routine written in Python to remove data entry errors, misspellings and inconsistencies. Next, each of the three algorithms were implemented using the Weka machine learning system: Naive Bayes Classifier (NBC), K-Nearest Neighbor (KNN) and Support Vector Machine (SVM). During the supervised training phase, each algorithm was trained to associate correct pairs of input text (job descriptions) and output results (job titles). After that, for each algorithm, a test phase with a different subset of job ads were tested to see how well each algorithm had learned to recognize job descriptions associated with specific job titles. These cycles of training and test phases were repeated 10 times, with different subsets of test and training sets for the job ads for each cycle to perform a K-fold Cross Validation.

Accuracy is defined as the number of correctly predicted results (true positives and true negatives) in proportion to the total observations. Precision is a measure of positive predictive value, while recall calculates the proportion of relevant instances that have been retrieved in relation to the total number of relevant instances. F1 is the weighted average of precision and recall, and thus takes both false positives and false negatives into account (Powers 2011). The classification accuracy and f1-scores of each of the three algorithms are shown below in Table 1.

Table 1. Accuracy and F1 of classifier algorithms

Algorithm	Accuracy	F1
Naïve Bayes Classifier	0.78	0.77
K-Nearest Neighbor	0.72	0.71
Support Vector Machine	0.91	0.91

Although the naive Bayes algorithm would be considered a relatively simple statistical method, it performed slightly better than the K-Nearest Neighbor algorithm, which performed poorest. The performance of the Support Vector Machine was the best of all three algorithms. These findings are similar to the results reported by Elnahrawy (Elnahrawy 2002). He found the naive Bayes classifier simple and fast, both in learning as well as in classification. Significantly, the naive Bayes classifier outperformed the K-Nearest Neighbor in both speed and accuracy. The Support Vector Machine delivered excellent results with respect to accuracy. The disadvantage was that it was very slow to learn and thus computationally expensive. Based on these findings Support Vector Machine (SVM) was selected to perform the classification of the job ads.

The next step was to classify job ads according to job titles. The most frequently identified job titles found during the test phase are shown in Figure 2.

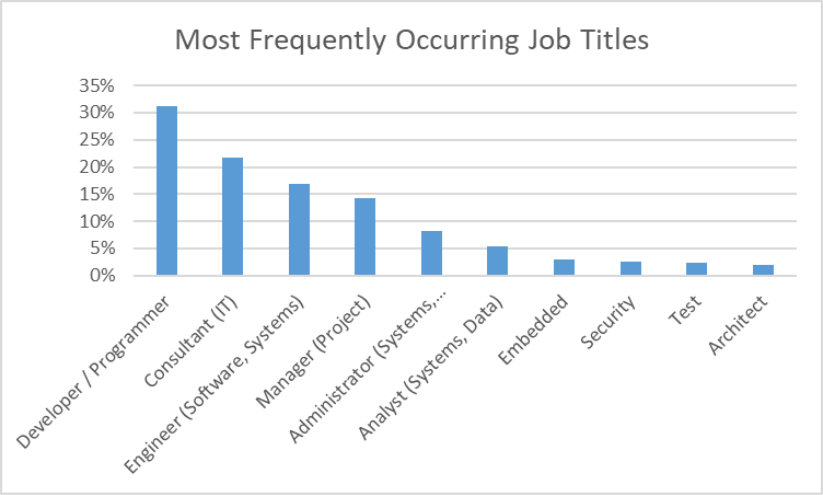

Figure 2. Most frequently identified job titles

These results provide an initial starting point to identify the most promising career opportunities for IT graduates. As expected, technically-oriented job titles, such as developers, programmers and software engineers appeared quite frequently. Interesting to note is that a significant number of highly sought job titles, such as consultant, analyst and project manager, are fields which are not limited to programming.

The next step was to identify the key competencies desired by prospective employers. Here, an unsupervised K-means clustering algorithm, as described in Section 3 Methodology, was used to group similar terms. Terms were grouped together in to a number "K" of clusters. Some of the clusters which formed were not very helpful. For example, one of the clusters formed included the words 'status', 'gender', religion', 'race' and 'disability'. It can be inferred that these terms are part of standard non-discrimination clauses. Especially of interest here are the clusters related to technical and non-technical skills. The most common clusters are displayed in Figure 3, along with the percentage of job ads which contained these terms.

As expected, job offers contained a number of specifications defining desired technical competencies. Computer programming skills and knowledge of software engineering, from initial requirements engineering, design, development, test and integration of software systems remain core competencies of any computer science degree program.

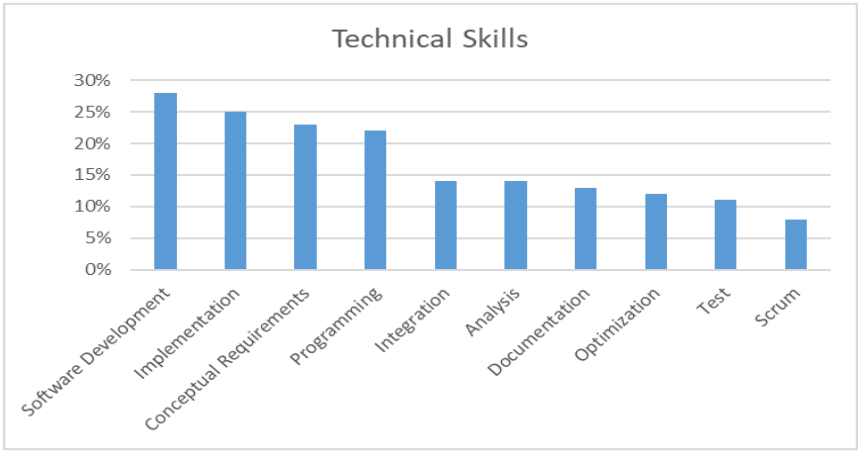

Figure 3. Most commonly sought technical skills

Significant for this research is that a number of non-technical skills were also specified as highly desirable by prospective employers.

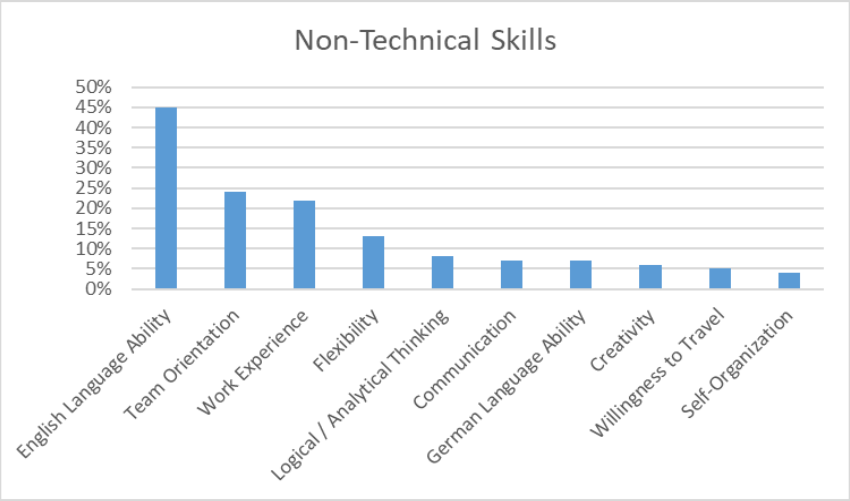

Figure 4. Most commonly sought non-technical skills

The ability to communicate effectively, both in the native language (German) but especially in English, were the most common non-technical skills sought. Working effectively in teams, flexibility, logical and analytical thinking, creativity and self-organization highlight abilities which would be considered soft skills. The willingness to travel was also specified as desirable for some careers.

5. CONCLUSIONS AND FUTURE WORK

In conclusion, the most commonly advertised job titles, technical and non-technical competencies for computer science graduates have been identified in a university career database. Machine learning methods proved effective in extracting useful information from large amounts of unstructured data. The ramifications of the knowledge gained, as presented in Section 4 Results, have consequences for underrepresented students and for the computer science curriculum at universities. It has been demonstrated that employers value both technical skills as well as personal characteristics when recruiting IT graduates. Although project management is often a mandatory course, the development of soft skills tends to be overlooked. Team training and conflict management are rarely taught explicitly in undergraduate computer science programs.

Students from underrepresented groups may be able to contribute unique soft skills, which traditional students may lack. For example, the ability to plan and organize multiple competing tasks can be vital for the success of large software development projects. Working together in teams often requires learning to resolve conflicts between team members. The ability to communicate both verbally as well as in written form is often not taught explicitly as part of the curriculum in computer science. Women tend to score higher in verbal ability, while men tend to score higher in spatial ability (Lewin 2001). This could imply that women may have an inherent advantage in communication skills. For students from migration and non-academic backgrounds, however, written skills may prove especially challenging. Extra support in learning how to improve their written skills, especially in their non-native language, may be necessary. Growing up bilingual and bi-cultural enables someone to build bridges between stakeholders and team members from different countries. The importance of bi-cultural proxies as bridge builders is discussed in MacGregor, et al. (MacGregor 2005). Bi-coded individuals are defined as people, who due to their life history, are able to operate equally well in two different cultures. Such bi-coded individuals can help serve as an intercultural translators between two cultures.

As opposed to concentrating on the disadvantages underrepresented students have, diversity can contribute unique advantages. Woolley, et al. (Woolley 2010) examined the role of collective intelligence performance. They found that diversity within teams tends to increase the collective intelligence of the entire team significantly (Woolley 2015). Bear, et al. (Bear 2011) found that diverse teams showed increased participation and collaboration, which led to a higher perception of efficacy among team members.

Especially groups performing creative or innovative tasks tend to benefit more from diversity (Williams1998).

In conclusion, students from underrepresented groups represent a potential source of untapped talent and could contribute to a diversity of perspectives in computer science. Future research will focus on evaluating in detail how the specific non-technical soft skills desired by prospective employers can be fulfilled by students from underrepresented groups. Interviews, surveys and machine learning analysis of text communication artifacts will be conducted with students from a migration background, first generation students, single parents and female students, to ascertain which unique advantages they can bring, especially in soft skills gained during their life history. This work is part of a larger research project to recruit, support and retain students from underrepresented groups in STEM subjects, as described in (Schuhbauer 2019).

ACKNOWLEDGEMENT

This research was supported in part by a grant from the Staedtler Trust for the research project “DiaMINT”.

REFERENCES

BAM 2011, Bundesanstalt fuer Migraiton und Fluechtlinge, Bestandsaufnahme und Vernetzung, *Vernetzungsworkshop: Integration von Studierenden mit Migrationshintergrund an deutschen Hochschulen*.

Basu, A., Walters, C. and Shepard, M., 2003, Support vector machines for text categorization, *HICSS 2003 Proceedings of the 36th Hawaii International Conference on System Sciences*, Big Island, HI, U.S.A.

Bear, J. B. and Woolley, A. W., 2011, The role of gender in team collaboration and performance, *Interdisciplinary Science Reviews*, Volume 36, Number 2, pp. 146-153.

Beecham, S., Clear, T., Barr, J., Daniels, M., Oudshoorn, M. and Noll, J., 2017, Preparing Tomorrow's Software Engineers for Work in a Global Environment, *IEEE Software*, Number 1, Vol. 34, pp. 9-12.

Branz, L., Pastran Reina, L., Richter, J., Waizmann, B. and Brockmann, P., 2019, Sentiment Analysis of Male and Female Developer Comments: Exploring Gender Influence on Emotinal Expressions in Software Engineering Projects, *GE@ICSE19 Second Workshop on Gender Equality in Software Engineering,* Montreal, Canada.

Budinska, L. and Mayerova, K., 2017, Graph Tasks in Bebras Contest - What does it have to do with gender*?, CSERC 17 Computer Science Education Research Conference*, Helsinki, Finland.

Chen, J., Huang, H., Tian, S. and Qu, Y., 2009, Feature Selection for text classification with Naive Bayes, *Expert Systems with Applications*, Number 36, pp. 5432-5435.

Cheryan, S., Siy, J., Vichayapai, M., Drury, B. and Kim, S., 2011, Do Female and Male Role Models Who Embody STEM Stereotypes Hinder Women’s Anticipated Success in STEM?, *Social Psychological and Personality Science*, Volume 2, Number 6, pp. 656-664.

Elnahrawy, E., 2002, Log-Based Chat Room Monitoring Using Text Categorization: A Comparative Study, *Proceedings of the International Association of Science and Technology for Development Conference on Information and Knowledge Sharing*, St. Thomas, U.S. Virgin Islands.

Faulkner, W., 2009, Doing gender in engineering workplace cultures. Observations from the field, *Engineering Studies*, Volume 1, Number 1, pp. 3-18.

Hinze, A., Timpany, C., Bowen, J., Chang, C., Starkey, N., and Elder, H., 2018, Collecting Sensitive Personal Data in a Multi-Cultural Environment, *HCI 2018 Proceedings of the International Human Computer Interaction Conference*, Las Vegas, Nevada, U.S.A.

Hoda, R., Ali Babar, M., Shastri, Y., Yaqoob, H., 2016, Socio-Cultural Challenges in Global Software Engineering Education*, IEEE Transactions on Education*, Number 3, Volume 60, pp. 173-182.

Huang, A., 2008, Similarity Measure for Text Document Clustering, *New Zealand Computer Science Research Student Conference,* Christ Church, New Zealand

Ilumoka, A. 2012, Strategies for overcoming barriers to women and minorities in STEM, *IEEE 2nd Integrated STEM Education Conference (ISEC)*, Ewing, New Jersey, U.S.A., pp. 1-4.

Inthn 2018, Technical University of Applied Sciences Nuernberg Computer Science Department, Lehrbericht 2018.

Jain, A., 2010, Data Clustering 50 Years Beyond K-Means *Pattern Recognition Letters*, Number 31, pp. 651-666.

Korde, V., 2012, Text Classification and Classifiers: A Survey, *International Journal of Artificial Intelligence and Applications*, Number 2, Volume 3, pp. 85-94.

Lewin, C., Wolgers, G., Herlitz, A., 2001, Sex differences favoring women in verbal but not in visuospatial episodic memory, *Neuropsychology*, Volume 15, Number 2, pp. 165-173.

Liebenberg, J. And Pieterse, V., 2016, Career Goals of Software Development Professionals and Software Development Students, *CSERC 16 Computer Science Education Research Conference*, Pretoria, South Africa, pp.22-28.

MacGregor, E., Hsieh, Y. and Kruchten, P., 2005, Cultural Patterns in Software Process Mishaps: Incidents in Global Projects, *HSSE 05 Proceedings of the 2005 workshop on Human and social factors of software engineering*, St. Louis, Missouri, U.S.A.

Powers, D. M. W., 2011, Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation, *Journal of Machine Learning Technologies*, Volume 2, Number 1, pp. 33-37.

Rish, I., 2001, An empirical study of the naive Bayes classifier, *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, Seattle, Washington, U.S.A., pp. 41-46.

Schuhbauer, H. 2019, Evaluation of an E-Learning Course in Moodle in Comparison to a Traditional Lesson, ICICTE2019 Proceedings, 19th International Conference on Information Communication Technologies in Education, http://icicte.org/assets/icicte2019proceedings.pdf, pp. 44-53.

Sebastiani, F., 2002, Machine Learning in Automated Text Categorization, *ACM Computing Surveys*, Volume 34, Number 1, pp. 1-47.

Speer, R.,2019,ftfy, Version 5.5, doi = 10.5281/zenodo.2591652, https://doi.org/10.5281/zenodo.2591652.

Technical University of Applied Sciences Nuernberg, 2004, Job-Boerse, https://www.hochschuljobboerse.de, lastaccessed April 26, 2019.

Technical University of Applied Sciences Nuernberg, 2018, Diversity an der Hochschule, https://www.th-nuernberg.de/hochschule-region/strategie-und-profil/hochschule-der-vielfalt, lastaccessed = April 26, 2019.

Tong, S., Koller, D., 2001, Support Vector Machine Active Learning with Applications to Text Classification, *Journal of Machine Learning Research*, Volume 2, pp.45-66.

Williams Woolley, A., Aggarwal, I. and Malone, T. W., 2015, Collective Intelligence and Group Performance, *Current Directions in Psychological Science*, Volume 24, Number 6, pp. 420-424.

Williams Woolley, A., Chabris, C. F., Pentland, A., Hashmi, N., and Malone, T. W., 2010, Evidence for a collective intelligence factor in the performance of human groups, *Science*, Volume 330, Number 6004, pp. 686-688.

Williams, K. and O'Reilly, C., 1998, Demography and diversity in organizations: A review of 40 years of research, *Research in Organizational Behavior*, Volume 20, pp. 77-140. bb

Williams, K. Y., & O'Reilly, C. A. I. 1998, Demography and diversity in organizations: A review of 40 years of research, In L. L. Cummings (Ed.), *Research in organizational behavior,* Vol. 20, pp. 77–140.

Wong, T., 2015, Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation, *Pattern Recognition*, Volume 48, pp. 2839-2846.