Diganta Rashed and Logan Roach   
IST 736 – Text Mining: Final Project

Automated Categoric Resume Screening: a KNN/LDA model approach.

**Introduction**

In the advent of modern-day automated hiring models, algorithmic processing of resumes is an all-too-common practice. These methods and models have significantly transformed the recruitment landscape, offering both opportunities and challenges to organizations and job seekers alike. These methods aim to streamline the hiring process, reducing the time and resources spent on sifting through countless resumes to identify the most promising candidates. This process of automation often primarily relies on machine learning models to optimize the process. Among these models, Latent Dirichlet Allocation (LDA) and K-Nearest Neighbor (KNN) models are two models to consider, for their capacity to revolutionize resume screening, offering nuanced insights that go beyond traditional keyword matching.

Latent Dirichlet Allocation is a machine learning algorithm that excels in discovering the underlying topics present in a collection of documents. By analyzing the text within resumes, LDA can identify patterns and themes that represent the candidate's skills, experiences, and professional interests. This capability allows hiring managers to understand the depth and breadth of a candidate's expertise in a way that keyword searches cannot. The K-Nearest Neighbor model, on the other hand, offers a complementary approach by focusing on the similarity between candidates. When a resume is added to the set, KNN evaluates it in the context of existing resumes that have been tagged as successful hires. By identifying the most similar elements within the resume pool, the model can predict the likelihood of a candidate's success in each role. This method is particularly useful for roles where the requirements are well-defined and consistent, as it allows for the comparison of candidates based on real-world outcomes. Furthermore, when used in conjunction with LDA, KNN can refine its predictions by considering thematic similarities between candidates, leading to a more nuanced understanding of fit beyond mere qualifications or experience.

This project attempts to replicate the approach of both models and stage a comparative analysis for the two models, using a set of resumes from various industries and categories, to understand the approach for each and compare pros and cons. To screen a diverse set of 3446 resumes, we will utilize our LDA and KNN models in a two-step process. First, LDA will analyze the resumes to identify underlying themes, categorizing candidates by skills and expertise. This thematic analysis allows for a deeper understanding of each applicant's professional background. Thena KNN model is used to compare a training set of resumes against a testing set to find the most similar profiles. By combining LDA's thematic insights with KNN's similarity-based predictions, we aim to streamline the screening process and observe the behavior of the two models in streamlining the resume screening process.

**Methods**

Our approach to creating the two models involves numerous functions and operations, but primarily relies on the sklearn, nltk, PyPDF2, wordcloud, numpy, pandas, seaborn and os libraries. Additional single-use case functions are also used to supplement the primary functions from the major libraries. Initial data cleaning and preprocessing begins with concatenating the data samples and reading the data in through two separate steps for data uniformity. First, the csv file containing a list of the resumes is read in and visually displayed to ensure proper reading in our data frame. Next, to observe the structure of our data, a pie chart for categories as well as the top 10 rows of the data frame is printed to observe and analyze the elements associated with the data. Word clouds were also used to explore the top words found in the resumes with respect to each category. The text from each resume labeled by category is instantiated with a TF-IDF Vectorizer which step tokenizes, counts and computes the TF-IDF vectors for the resumes. The vectorized resumes are used as features and the resume categories as the target variables. The data is split into training and test sets where 80% of the data is used for training the model and the rest is used to test the model for performance. The training and test sets are then used for each of our models of interest. The KNN model is evaluated using accuracy scores and a classification report including the precision, recall, and F1 scores. With respect to the LDA model, the fitted LDA model is used to assign topics to each resume based on the probability distribution of topics. The topic distributions are used as features for building a categorization model, in the first case a Support Vector Classifier was used and the second was a Random Forest classifier. The performance of these models is evaluated using the accuracy of the model with and without the LDA model topics as features to determine if using the LDA model topics as features can improve the classification of the resumes when using more traditional techniques.

**Results**

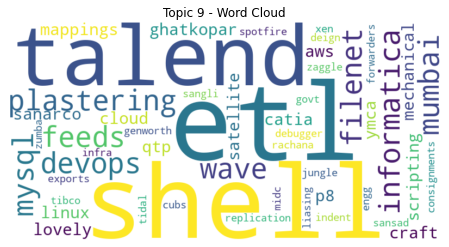
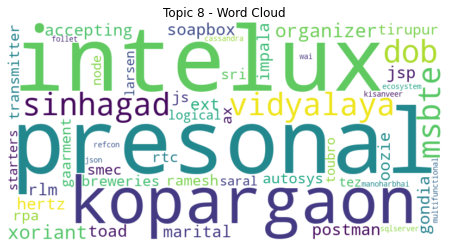
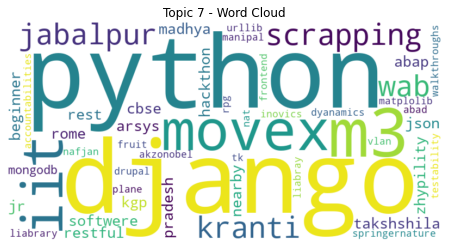
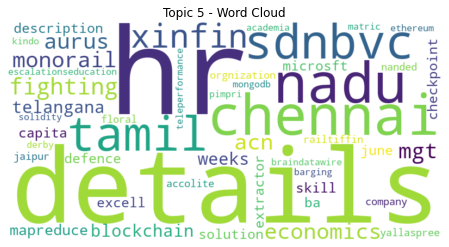
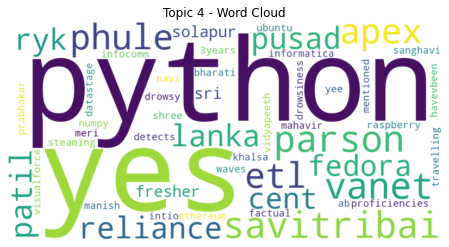
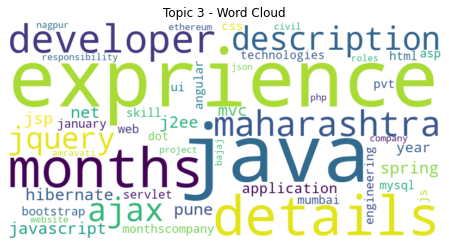
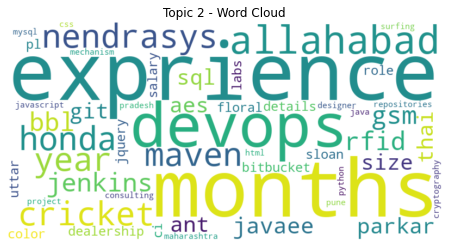
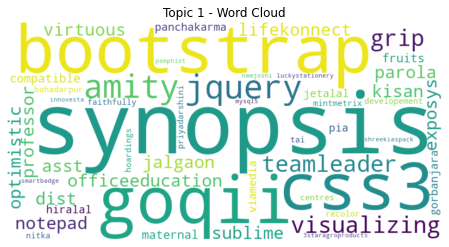
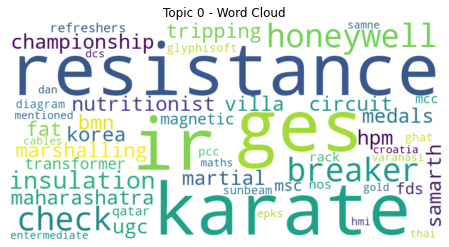
Results of the KNN-Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category | Precision | Recall | F1-Score | Support |
| ACCOUNTANT | 0.5 | 0.67 | 0.57 | 24 |
| ADVOCATE | 0.5 | 0.46 | 0.48 | 24 |
| AGRICULTURE | 1 | 0.15 | 0.27 | 13 |
| APPAREL | 0.62 | 0.26 | 0.37 | 19 |
| ARTS | 0.6 | 0.29 | 0.39 | 21 |
| AUTOMOBILE | 0.67 | 0.29 | 0.4 | 7 |
| AVIATION | 0.78 | 0.3 | 0.44 | 23 |
| ADVOCATE | 1 | 0.75 | 0.86 | 4 |
| ARTS | 1 | 1 | 1 | 7 |
| AUTOMATION TESTING | 1 | 1 | 1 | 5 |
| BANKING | 0.6 | 0.39 | 0.47 | 23 |
| BPO | 0 | 0 | 0 | 4 |
| BUSINESS-DEVELOPMENT | 0.33 | 0.62 | 0.43 | 24 |
| BLOCKCHAIN | 1 | 1 | 1 | 8 |
| BUSINESS ANALYST | 1 | 0.83 | 0.91 | 6 |
| CHEF | 0.79 | 0.62 | 0.7 | 24 |
| CONSTRUCTION | 0.63 | 0.77 | 0.69 | 22 |
| CONSULTANT | 0.43 | 0.13 | 0.2 | 23 |
| CIVIL ENGINEER | 0.71 | 1 | 0.83 | 5 |
| DESIGNER | 0.65 | 0.62 | 0.63 | 21 |
| DIGITAL-MEDIA | 0.59 | 0.53 | 0.56 | 19 |
| DATA SCIENCE | 0.86 | 0.75 | 0.8 | 8 |
| DATABASE | 1 | 1 | 1 | 7 |
| DEVOPS ENGINEER | 1 | 1 | 1 | 11 |
| DOTNET DEVELOPER | 0.67 | 1 | 0.8 | 6 |
| ENGINEERING | 0.68 | 0.62 | 0.65 | 24 |
| ETL Developer | 1 | 1 | 1 | 8 |
| ELECTRICAL EGINEERING | 1 | 1 | 1 | 6 |
| FINANCE | 0.56 | 0.58 | 0.57 | 24 |
| FITNESS | 0.55 | 0.48 | 0.51 | 23 |
| HEALTHCARE | 0.42 | 0.57 | 0.48 | 23 |
| HR | 0.67 | 0.77 | 0.72 | 31 |
| HADOOP | 1 | 1 | 1 | 8 |
| HEALTH AND FITNESS | 1 | 1 | 1 | 6 |
| INFORMATION-TECHNOLOGY | 0.57 | 0.71 | 0.63 | 24 |
| JAVA DEVELOPER | 0.81 | 1 | 0.89 | 17 |
| MECHANICAL ENGINEER | 1 | 1 | 1 | 8 |
| NETWORK SECURITY ENGINEER | 0.83 | 1 | 0.91 | 5 |
| OPERATIONS MANAGER | 0.8 | 1 | 0.89 | 8 |
| PMO | 1 | 1 | 1 | 6 |
| PUBLIC-RELATIONS | 0.52 | 0.68 | 0.59 | 22 |
| PYTHON DEVELOPER | 1 | 1 | 1 | 10 |
| SALES | 0.33 | 0.52 | 0.41 | 23 |
| SAP DEVELOPER | 0.71 | 1 | 0.83 | 5 |
| SALES | 1 | 1 | 1 | 8 |
| TEACHER | 0.55 | 0.8 | 0.65 | 20 |
| TESTING | 1 | 1 | 1 | 14 |
| WEB DESGINING | 1 | 1 | 1 | 9 |
| Accuracy |  |  |  | 0.65 |
| Macro Avg | 0.75 | 0.73 | 0.72 | 690 |
| Weighted Avg | 0.67 | 0.65 | 0.64 | 690 |

The KNN model provided us with a set of precision, recall, F1 score and support scores for each category, training over the set of resumes in each category and providing us with probabilistic results. An assortment of categories, such as Testing, Web Designing, Python Developer, Database and DevOps engineer provide us with perfect Precision and Recall scores, implying that the samples have consistent, yet the accuracy scores only show high performance on the notably smaller samples. In the overall results, our model shows a 67% precision and 65% recall, implying low false positives and moderate to high success in accurate recall over its iteration through the testing set of our data, which, when split, included a total of 690 resumes identified.

The LDA found ten topics to apply to the previously vectorized resume texts. The topics can be found below in the table and represented visually in word clouds.

Word Clouds of Topics Found Using the LDA Model



The topics in the LDA model were used as features for building the two different categorization models. The number of topics used in the LDA model iterated across 10, 100, 500 and 1000 topics. When training the Support Vector Classifier (SVC) with the LDA determined features, the accuracy was found to be 0.1174 or 11.74%, an abysmal performance, and when using an SVC without the LDA model topics as features the accuracy increased to a more respectable 0.7391 or 73.91%. The number of topics used in the LDA model iterated across 10, 100, 500 and 1000 topics. When training the Support Vector Classifier (SVC) with the LDA determined features, the accuracy was found to be 0.1174 or 11.74%, an abysmal performance, and when using an SVC without the LDA model topics as features the accuracy increased to a more respectable 0.7391 or 73.91%. The previous results used an LDA model with 10 topics, when increasing the model topics to 500, there were some improvements when using the topics as features, 27.97% and 53.76% for the SVC and Random Forest model respectively, but nothing reaching the baseline of using the classification models traditionally without the LDA topics as features. The other iterations of the LDA model with 100 topics (25.07% accuracy for the SVC, 45.22% accuracy for the Random Forest), and 1000 topics (14.2% accuracy for the SVC, 32.75% accuracy for the Random Forest) did not perform as well when using 500 topics.

**Conclusion**

Applying the LDA model topics as features when building the categorization models did not turn out to be a fruitful endeavor. Using the models in a more classical analysis, simply using the vectorized data without incorporating the LDA topics, would be a more efficient use of one's time. Even if the data was processed differently, and the LDA model was tuned differently, the results when using the model topics as features were so poor, any endeavor would be better suited using the models traditionally. Thus, tunning, adjusting the data processing, or parameters of the Support Vector Classifier (SVC) and Random Forest model using the vectorized data would be the most beneficial for improving the classification of the resumes. Additionally, another potential leverage method that could have been recognized was training the KNN model over the LDA model, which would provide us with a point of reference and allow for stronger accuracy for the LDA as well as KNN model.

Some sources of improvement primarily lie in the stronger and further optimization of the LDA model, and it lies in finding the right stop words for the program. Increasing the number of stop words may provide us with an output that would grant us the most coherent set of words, not only for the word clouds visibly, but also for the KNN model to be accurately trained over. An increased data set can also greatly improve the model's ability to train over the set of resumes and provide us with a more effective filtration of the resumes.