Employment Recommender System Using Natural Language Processing

By

Muhammad Azkaenza, Erik Konstenius,

Joachim Elias Aslaksen Jonasson, Danielle Verniece Duncan

Student Numbers: 150419, 149880, 14988, 142892

Course: Text Analytics (KAN-CDASO2030U)

Degree: MSc in Business Administration and Data Science

Submission Date: May 30, 2022

Supervisor: Raghava Rao Mukkamala

14.8 pages or 32.873 characters

ABSTRACT

Finding a job is a challenging process for a lot of people. While some have an idea of what job paths are open to them, many do not know all the types of jobs and job titles that they are qualified for. In this paper, an alternative to the simple content-based recommender system was proposed that incorporates topic modelling to create recommendations within a set of topics that would best fit the user. This system helps the user understand which fields they are qualified for and what jobs are currently available in those fields that match the user's experience. The results demonstrated in this paper show that incorporating topic modelling can significantly reduce the noise caused by individual terms and that the system can generate accurate recommendations on CVs with a varied background.

KEYWORDS

Content-based recommender system, topic modeling, Top2Vec, job search, machine learning, resume

TABLE OF CONTENTS

Introduction

Motivation

A common technique in traditional content-based recommender systems is to recommend items based on a query of the entire corpus. This paper investigates whether limiting the recommendation to a subset of the corpus that is the most relevant to the user using topic modelling can yield better recommendations. The recommender system proposed in this paper has been applied to generate job recommendations based on a user's CV. Seeing that jobs belong to particular fields, we believe that it is motivated to investigate if incorporating topic modelling can result in better recommendations.

Research Question

Can topic modelling improve the recommendations of a content-based recommender systems within the application of job recommendations?

Related Work

With the rise of the Internet in the early 1990s, different methods for retrieving and searching for information became prevalent as documented in the paper by Al-Otaibi & Ykhlef (2012). As the number of users and the amount of information available on the Internet was rising rapidly, it became apparent that filtering and producing efficient methods for showing the most relevant information to users was important. Hence recommender systems came to fruition during this period as a way of showing relevant information to users. Since then, it has become a relevant topic and a useful application for a plethora of use cases.

Recommender systems are implemented in situations where a user needs to quickly process large amounts of data where only a subset of the data is considered valuable for the user. The goal of such a system is to produce a model that can help the user to make faster or better decisions. An example of a simple recommender system (document/corpus via cosine similarity) is used as the baseline model in this paper.

Bansal, Srivastava & Arora (2017) conducted a research paper where the aim was to produce an efficient and accurate content-based recommendation engine that matches users with relevant jobs. Their recommendation engine matches the skills and interests of users with the features of job postings. The authors use real world data of job postings and implement LDA to the job roles to extract topics based on job roles from the user profile. After the topics are found through LDA, frequency and parts of speech filters are applied to remove weaker and less relevant features. In their conclusion they explain that their recommendation engine is performing better than traditional models due to the implementation of LDA (Bansal, Srivastava & Arora., p. 865-872, 2017).

Mishra & Rathi (2020) evaluate different job recommender systems by comparing three different datasets of job recommendation systems from LinkedIn, work4, and CareerBuilder and evaluate their performance with three algorithms. The algorithms comparing performance were Support Vector Machine, a graph-based approach and K-Nearest Neighbor. The graph-based approach showed optimal results for recommending jobs when tested on all three datasets (Mishra & Rathi., p. 9-10, 2020).

Conceptual Framework

Web scraping/crawling

In 1993, the first web crawler was developed. It was initially used for measuring the size of the Web, but later on, it was used to retrieve web pages information given a URL (Abu Kuasa, et. al, 2013). Manning, et. al (2009) defined web crawling as a process of getting and indexing web pages to help search engines quickly and effectively give results to user queries. It serves as a support for search engines in providing responses to users in a timely manner. On a similar note, Ceri, et. al (2013) explained that web crawling is defined as content exploration activities. It is performed automatically in identifying, analyzing, and cataloguing web pages to retrieve information and interconnectivity between pages. This exact function of a web crawler was used in collecting data for this project. Technically, the web crawler took a URL as input, then analyzed and indexed all resources available through the page. Before downloading, the web crawler allowed the user to tailor the data collection workflow according to specific web page structure and user specific needs. This is explored further in the Data Acquisition Section.

Recommender systems

Collaborative filtering and content-based filtering are two of the most common forms of recommender systems. In collaborative filtering, recommendations are generated from what similar users appear to like. An example could be how a customer in a store receives an advertisement that is tailored to what demographic the customer belongs to and how that demographic behaves. Content-based recommender systems generate recommendations through past experiences, like they can suggest movies to watch based on recently watched movies (Lops et al., 2019). Content-based systems are unable to capture quality or what other people appreciate in the recommendations. It simply recommends similar objects to what the user has already experienced (Lops et al., 2019). This implies that the system can also generate recommendations that fit a very small group of users (Google Developers, 2018). Content-based recommender systems can cause overspecialization where the user is never recommended objects that the user has not experienced before. A good model is said to have a certain level of novelty and serendipity where the user is shown objects that it has not experienced before but could think is interesting (Aggarwal, 2016). Collaborative filtering on the other hand can detect trends and popularity between objects (Lops et al., 2019). Collaborative filtering suffers from the cold-start problem where objects not seen during the training are not easily integrated into the recommendations (Google Developers, 2018). New objects are not a problem for content-based systems as already existing objects can be compared to the new object.

Today, many of the most powerful recommendation systems use both collaborative filtering and content-based filtering to generate recommendations. It is also common to incorporate surrounding or external data that could help make even more relevant recommendations (Lops et al., 2019). Research shows that corporations and research institutions rarely use content-based systems without incorporating external data or other recommender systems (Aggarwal, 2016).

Evaluating Topic Models

Evaluating a topic model is not always an easy and straightforward task. When a model is used for a quantitative task, such as classification, calculating its effectiveness can be rather simple. However, when the model is utilized for a qualitative task, such as examining semantic-relation in the content-based recommendations, evaluation becomes more challenging.

Evaluation methods, such as human observation and interpretation, have always played a large role in interpreting the results of topic modelling. Unfortunately, this method is sometimes impractical, time-consuming, and expensive as these methods highly rely on human capabilities in making judgments and often require domain-expertise.

Quantitative approaches, such as perplexity and coherence, are used to complement the limitation of human qualitative evaluation. Perplexity is used to measure the quality of a model-independent from any application which evaluates the ability of a topic model to predict a test set after being trained (Jurafsky & Martin, 2021). Yet, according to Chang, et al. (2009) it fails to measure the relationship between words, topics, and documents. Roeder et al. (2015) introduces a model to quantify the coherence relationship of topics modeling which laid the foundation of the coherence model in the Gensim package. It consists of four stages: (1) segmentation of word subsets which involves choosing how words are grouped together, (2) probability estimation, (3) confirmation measures the relation among word grouping, and (4) aggregation is the summary of the measures, resulting in a coherence score.

The methodology used in this paper to evaluate topic modeling technique is designed to be independent of our own ability to make good estimations of what is a well performing technique. We developed a simple measure that could be compared between all techniques. The formula is shown below.

$$Performance\ score\ =\ \frac{\#\ of\ topics\ that\ can\ be\ labelled\ in\ an\ understandable\ way}{Total\ number\ of\ topics}$$

This technique requires manual labeling of each generated topic. A low score could mean that the technique generates too few topics causing too generic and meaningless topics to form. It could also be that the model generates too many topics whereby a large portion of the topics are not relevant. A third interpretation of a low score could be that the technique is performing poorly. The last case could be because of a too simple model or by insufficient preprocessing. Examples of insufficient preprocessing could be if stop words have not been removed while also not using TF-IDF. A high score on the other hand would mean that the model is not only able to pick up separable and understandable topics, but also that a large portion of all topics are separable and understandable. The performance score will be used together with the coherence score.

Methodology

First, the data containing job listings and CVs were cleaned. The topic modelling methods LDA, NMF and Top2Vec were used to generate topics, or in this case fields, from the dataset of job descriptions. The topics in the best performing method were then labelled and the topics from the other method were discarded (see figure 1).

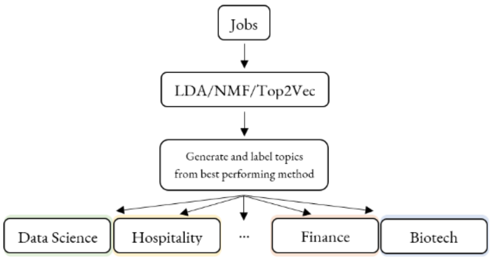


Figure 1

The three topics that were the most similar to the CV were found as well as the 250 jobs that were the most similar to each of the top three topics. For each topic now containing 250 of the most relevant jobs for that topic, we find the 20 jobs that are the most similar to the CV. These 20 jobs are the final recommendations in descending order of similarity score. A summary of the algorithm can be seen below (see figure 2 and 3).

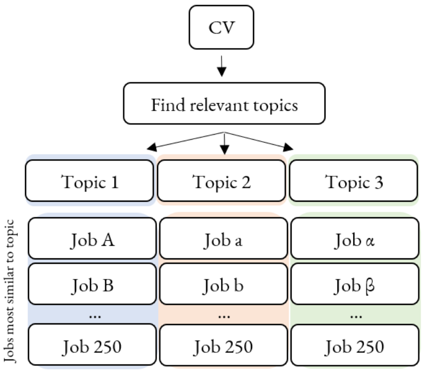


Figure 2

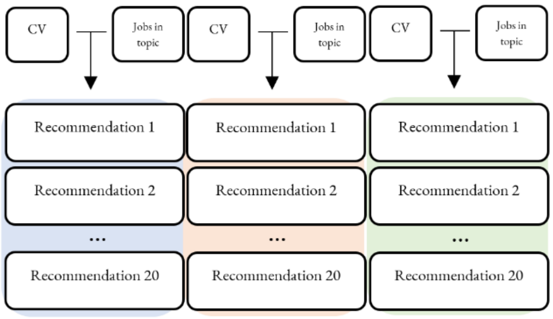


Figure 3

Data Acquisition

Job Listings

The job listings used in this study were retrieved from LinkedIn. LinkedIn was chosen because of its international popularity and broad offering of jobs. The data extraction was carried out by a custom-built web scraper using the software Octoparse. The scraper was given several queries that it processed. Each query returned data relating to a job listing. Figure 3 shows a simplified summary of the scraping process.

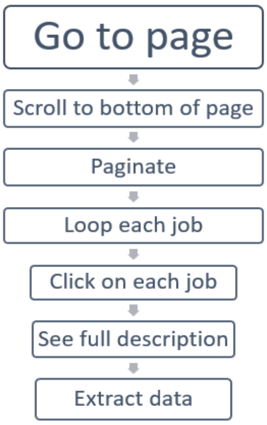


Figure 4

Some noteworthy complexities worth mentioning were:

- LinkedIn limits the search result per query to 40 pages of 25 job listings. Thus, the web-scraping needs to be done repeatedly, approximately every 1,000 job listings
- For every job listing scraped, the information needed to be saved from two different web pages. The title page with the title, image, link, location, date, and company; and the secondary details page consists of the detailed description, and various attributes.
- Since scraping was done from two separate pages the data could be wrongly scraped via mismatched page titles and details. This was either caused by an unstable connection from our end-user side or intentional disruption from LinkedIn to defend against scraping.

CVs

The CVSs used in this project for generating recommendation were samples from resumeviking.com. The script is designed to be able to process any type of CV if the file type is pdf. The full CVs can be seen in figure 1 and 2 in Appendix 1. The locations in the second CV were changed to European locations in order to limit the scraping to jobs in Europe.

Data Cleansing

Job Listings

Because data was scraped from 22 different searches the data was first joined, moved into a data frame, and then examined visually to determine potential holes or errors in processing. Holes found were corrected via moving misplaced data from other columns through manual selection. All job titles were analyzed through custom Regex to create more cohesive and similar job titles. Positions like "Data Analyst (45601) - English mand. in Munich office" were shortened to "Data Analyst". The library langdetect was used to identify and then remove all listings that were not in English. All listings with missing text were removed, line errors were corrected, and all text was converted to lowercase.

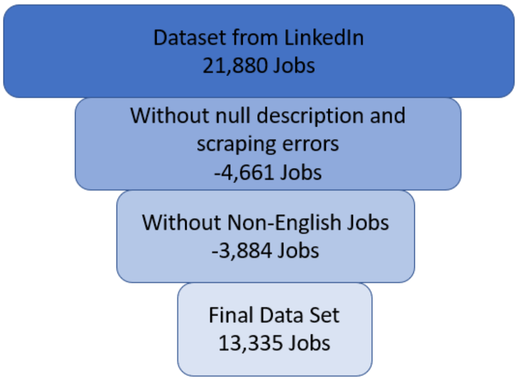


Figure 5

All text was then tokenized. Tokenization is the process of separating bodies of text into smaller pieces, referred to as tokens. Tokens allow for the creation of a vocabulary to limit the processing needed for models by adding parameters to the training set. Instead of using all of the English language or all possible letter combinations the vocabulary set can be limited to only words found in the tokens, thus creating a faster and more efficient program.

All URLs, special characters, and words shorter than two characters were also removed. Most text models make comparisons to the document and the corpus for words that occur within. Therefore, it is important to make sure that similar text is represented the same. This can be seen best with words like "who", "whose", "who's", and "who!". After processing the tokens all of these words were reduced to their root "who" using lemmatization for the most accurate representation in the model. This process is much more computationally expensive compared to other techniques, but it often generates better results that are also are more readable - which is essential for topic modeling.

CVs

The resumes were tokenized, and any non-alphabetical and short words were removed. The tokens were lowercased and lemmatized. Any stop words were removed. A community-generated list of stop words from NLTK's GitHub was combined with a self-generated list of words that are unique to resumes like "ability", "regional", and month names. The resulting list had better performance compared to NLTK's stopwords when the results were assessed manually.

Embedding

Vectorization converts text tokens into a numerical vector representation within the context of the corpus. This process can be done via extremely simple methods like assigning a number to all words in a dictionary and creating a vector with a count representation of the positions in the dictionary. This results in an extremely large and sparse (containing many 0s) vector that can be nearly impossible to extract data from.

The complexity of the representation of words is not only limited to the numerical representation but also how to represent the meaning and relationship of words. This moves beyond simple embedding methods into calculations of similarity, Euclidean space, and semantic space representations. Cutting edge work is moving away from the vector representation and into the multidimensional tensor application that can help to represent these complex relationships of words. These innovative methods like GloVe and word2vec are largely not used in this paper.

TF-IDF

Term Frequency (TF) - Term frequency is the number of times a specific term appears in each document divided by the sum of the words in the document. Sometimes, log-space is used to ensure that a word that appears a hundred times is not considered a hundred times more important than a word appearing just one time. The frequency is increased by one so terms that do not appear in the corpus get a weight of zero (Jurafsky and Martin, 2021).

$$tf(t,d) = \frac{f_{t,d}}{\sum f_{t',d}}$$

Equation 1

Inverse Document Frequency (IDF) - The inverse document frequency takes into consideration the frequency of the word in the entire corpus. Words like “is” or “and” are not as informative as other less frequent words. The denominator is called the document frequency and it shows how many documents a word appears in. The numerator shows the total number of documents in the collection. Log space is used to deal with large collections of documents (Jurafsky and Martin, 2021).

$$idf(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Equation 2

Term Frequency Inverse Document Frequency (TF-IDF) - The TF-IDF is the product of the term frequency and the inverse document frequency. The TF-IDF weight of a word in a document is given by the following formula (Jurafsky and Martin, 2021).

$$tf - idf(t,d) = tf(t,d) * idf(t)$$

Equation 3

Topic Modelling

The topic modelling techniques LDA, NMF, and Top2Vec were used to model topics in our dataset of jobs. The resulting topics were incorporated into the recommender system to make recommendations on a subset of jobs that fit into topics, or fields, that are relevant to the user.

NMF

NMF or non-negative matrix factorization is a simplified statistical approach to modelling data to find latent structures through factorization. NMF is a type of dimensionality reduction technique that is the “equivalent to low-rank matrix approximation” (Gillis, 2014). This is a very popular but highly simplistic model because the approach requires very little hyperparameter tuning and the results give “extremely interpretable factors” with an “intuitive and interpretable representation” (Gillis, 2014), (Turkmen, 2015). NMF may be simple but, according to Kim and Park (2008), “sparse NMF does not simply provide an alternative to K-means, but rather gives much better and consistent solutions to the clustering problem”. This allows NMF to achieve equivalent or better results with less processing time and power than more complex algorithms.

As Gillis outlines in his paper “The Why and How of Nonnegative Factorization”, NMF operates by generating a matrix (A or X) containing all documents by all the words in the vocabulary. More complex versions of NMF use embeddings, like TF-IDF, to allow for more information via scaling. This A/X matrix is factorized by taking the hyperparameter of topic count to generate a W (words in vocabulary by topic count) and H (topic by document) matrix. Because the basis elements of the W matrix are words found in multiple documents that allow them to be interpreted as topics and because the basis for the H matrix is the documents this allows us to see what documents belong to each topic. Therefore NMF “identifies topics and simultaneously classifies the documents among these different topics” (2014).

NMF was selected as a model to test for this content-based recommender system because of its outperforming K-means in sparse applications, fast run speed, and ease of understanding.

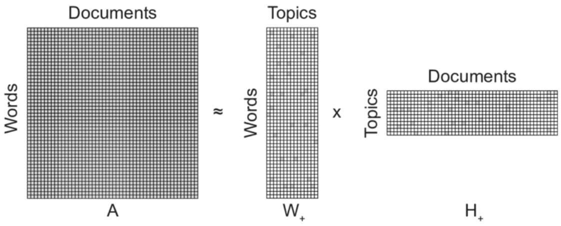


Figure 6 from Kuang, Brantingham, et al 2017

$$\underbrace{X(:,j)}_{j\text{th document}} \approx \sum_{k=1}^{r} \underbrace{W(:,k)}_{k\text{th topic}} \quad \underbrace{H(k,j)}_{\substack{\text{importance of } k\text{th topic} \\ \text{in } j\text{th document}}}$$

Equation 4 from Gillis 2014

LDA

LDA or Latent Dirichlet Allocation is a generative probabilistic model that utilizes a three-level hierarchical Bayesian model Blei, David M. et al (2003). This generative process allows the model to use "the imaginary random process by which it assumes the documents arose" (Blei, 2012). This imaginary process allows the model to assign a random distribution of words to a pre-specified number of topics and use the generative process to cycle through the words in the vocabulary to assign to different topics. LDA then cycles through documents and assigns statistical probabilities of the topics per document. This results in a three-layer hidden architecture of "the topics, per-document topic distributions, and the per-document per-word topic assignments" (Blei, 2012). This is where the LDA reverses the generative process to try and infer the hidden document structure by utilizing the Bayesian probabilities of the three layers. LDA has many tune-able hyperparameters that can change the number of topics, and control the document-topic density, the topic-word density, and the number of iterations.

Top2Vec

Top2Vec is the most advanced model used in this paper. Unlike the previous two models, Top2Vec does not require that the number of topics are defined. It is however able to group topics together if the initial model produces too many topics. Top2vec also utilizes semantic embedding to represent words via semantic association rather than relying on preprocessing like stemming, lemmatization, and text similarity. By applying Top2Vec using semantic space, it ensures that "big" and "large" are seen as very similar (Angelov, 2020). Top2Vec first creates a joint embedding of the document and word vectors by using doc2vec (an expanded word2vec model that is an "unsupervised framework that learns continuous distributed vector representations for pieces of texts" (Le & Mikolov, 2014).

This doc2vec embedding framework allows the model to capture "distributed word representations" via a neural network by utilizing a sliding window to learn and predict adjacent words within the text (Angelov2020). Top2Vec then uses these document/word joint embeddings to develop topic/document/word embeddings. These topic embeddings are developed via semantic space. As the doc2vec vectors are analyzed, the space of the document vectors (with the most semantically relevant words being most prevalent) the topics emerge as the dense areas of highly similar documents (Angelov, 2020). To find these dense areas within the sparse vectors, dimensionality reduction is performed on the vectors and then hierarchical density-based clustering is performed. After dense clusters are found and established the Top2Vec model uses semantic space to find the words closest to the generated topic vector and return those words as the representation of the topic.

Generating Recommendations through Similarity

Cosine similarity is used to calculate similarities and generate recommendations in this paper. The measure is an angle between two vectors, often used to find similarity between two vectors for different purposes. It is one of the most common metrics used specifically to measure the similarity between two specific words or documents as Jurafsky & Martin (2021) explains in chapter 6. In order to measure the similarity, the cosine angle is applied between two vectors of the same dimension. This method is more robust than applying a dot product between two vectors to measure similarity. In the case of using dot product, words that appear often have longer vectors and therefore will have similarities with other words that have longer vectors. The drawback with this method is that frequent words will have longer vectors and therefore have a higher dot product. This results in words appearing more frequently and having a higher similarity to each other, which is not true in many cases. To resolve this problem, a normalized dot product can be applied, which is the same as applying the cosine angle between two vectors. This method alters the dot product by dividing the dot product by the length of each of the two different vectors. The formula for calculating cosine between two documents is completed by the following formula, where v and w are two separate vectors.

$$v * w/|v| * |w| = cosine$$

Equation 5

The cosine value differs depending on direction, for example, a value of -1 means the two vectors are pointing in opposite directions. A value of 1 is for two vectors pointing in the same direction and for 0, the two vectors are orthogonal to each other (Jurafsky & Martin., ch. 6, 2021)

Two other prevalent techniques to calculate similarities include dice coefficient and Jaccard similarity (Bag, Kumar & Tiwari., p. 202, 2019). Both Jaccard similarity and dice coefficient are statistical methods used to find similarities between variables. Jaccard similarity can be used to find documents with similar text by measuring how close two sets are (Bag, Kumar & Tiwari., 2019). This method calculates the number of words overlapping in one document compared to another. It only uses words that are a part of both sets to create a ratio between them (Bag, Kumar & Tiwari., 2019). Dice coefficient is a similar technique, where the number of word tokens in both sets are counted and divided by the total number of word tokens from both sets (Sripada, Kasturi & Parai., p. 2, 2005). The result is a ratio of how similar the two sets are in terms of tokenized words.

Results

The first part of the results will look at how well the three mentioned topic modeling techniques (NMF, LDA, and Top2Vec) successfully distill meaningful topics from the retrieved corpus of job descriptions. The best performing technique will both be used for visualization purposes to capture similarities between topics as well as act as a component of the job-recommender system. The second part will evaluate how well the recommender system combined with topic modeling compares to the baseline model of a recommender-system.

Topic Model Selection

After all the topic models were run, they were assessed via methods highlighted in section 3.4. The assessment was based on their performance in topic modelling rather than job recommending due to the ability to generate comparable performance metrics between topics. Quantifiable performance metrics on job recommendations are difficult due to using real-life data.

Top2Vec

By running this model first, it provided a sense of scale for the number of topics needed for effective modeling. The first model generated 144 topics. A sample of the topics is shown in Appendix 2 figure 3 and 5. The results show that the model manages to largely segment the data into understandable and logical topics. These topics then needed to be hand labelled to be able to use them for job recommendations. 74 of the 144 topics were able to be segmented into understandable and logical topics. This means that the first Top2Vec model received a performance score of approximately 0.5139.

Using the information from the topics that were manually labeled in an understandable and logical way, topics were reduced to 70. The reduced Top2Vec contained 64 topics that could easily be labelled in an understandable and logical way. This means that it received a performance score of 0.9143. This is considerably better than the first model. The second model contains 13.5 % less labeled topics but does in total have more than 90 % of all its topics labeled.

LDA

Because LDA requires that the number of topics is specified an iterative process was carried out to find a good number of topics. The final LDA model contained 70 topics because it achieved good performance and could be directly compared to the Top2Vec with 70 topics. The 70 topic LDA model achieved a performance score of 0.27. A sample of topics is shown in Appendix 2 Figure 5. The model received the lowest score among all models. When assessing the model, it was found that the IT and finance-related topics were too general, meaning that software development and data science, for example, would be considered the same field. Since LDA uses Bayesian statistics to form probabilities for word occurrences it makes the model more sensitive to noise in the data. LDA also uses the bag of words approach so, unlike top2vec the sentence structure is not taken into consideration.

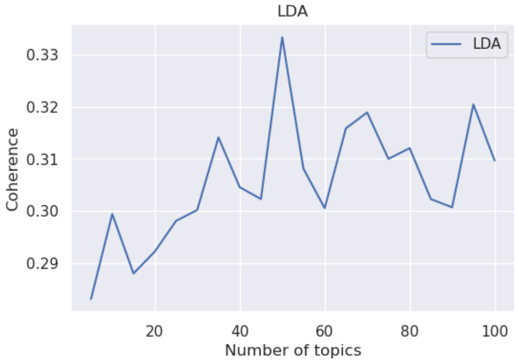


Figure 7: LDA Coherence Score

NMF

The same approach to finding the number of topics for the LDA model was performed for the NMF model which also resulted in choosing 70 topics. A sample of the topics is shown in Appendix 2 Figure 6. The model received a coherence score of 0.42 and a performance score of 0.47. This means that 33 out of a total of 70 topics could be easily recognized and labelled. The use of factorization to find structure and relation between words might be the reason why NMF performed better but still could not outperform top2vec.

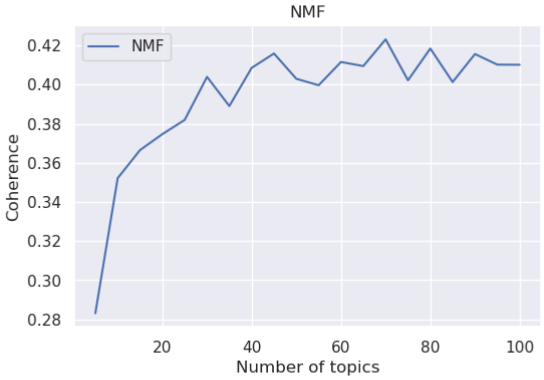
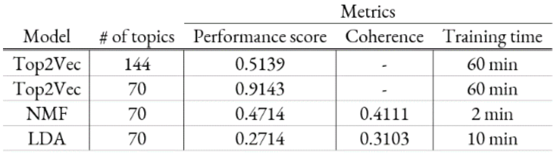


Figure 8: NMF Coherence Score

Conclusion of the topic modelling techniques

In conclusion, the Top2Vec with 70 topics received the highest performance score which implies it manages to create understandable and well-separated topics. It is, however, important to note that implementing a topic modeling technique into a recommender system that contains a lot of topics but receives a low-performance score can still perform well if the user's input, in this case, a CV is not like the unstructured and unlabeled topics. Otherwise, it will generate recommendations that do not follow any meaningful structure.

The Top2Vec with 70 topics will be used to build the job-recommender. A summary of all models is shown below.



		Metrics		
Model	# of topics	Performance score	Coherence	Training time
Top2Vec	144	0.5139	-	60 min
Top2Vec	70	0.9143	-	60 min
NMF	70	0.4714	0.4111	2 min
LDA	70	0.2714	0.3103	10 min

Figure 9: Summary of the performance of topic modeling techniques

Topics that receive the same name are grouped together and topics that were not labelled were dropped. The topics are then used to find the 250 most similar jobs to each topic. The recommendations are generated by calculating the similarities between the jobs for the topics that best fit the user and the CV.

EVALUATING RECOMMENDATIONS

The recommendations from the base model can be seen in figure 7 and 8 in Appendix 3. The person with the first CV receives almost solely recommendations from one of his previous employers "Johnson & Johnson". It appears as if the recommender system is not generating recommendations that are relevant to the person's skills and experience. The person with the second CV receives better recommendations as it seems to be able to pick up both her experience in aviation, hospitality and commerce. Yet, this model cannot generate recommendations on a subset of her CV.

When comparing the results, it can be observed that the proposed model appears to be able to reduce some of the noise that caused single terms such as "Johnson & Johnson" to receive a disproportionately high weight in the recommendations. The recommendations generated for the first CV are more relevant as they appear to be better at incorporating the person's experience. This is likely due to how the base model does not account for experiences in the same way as the model proposed in this paper. While the base model simply calculates similarities between the CV and all jobs, the model proposed in this paper calculates the similarities between a subset of all jobs that are considered the most relevant to the person. The jobs recommended within a topic do appear in multiple topics as seen in figures 9 and 10 in Appendix 3 for the first CV. An explanation could be that some topics are similar, and some overlap is expected. The overlap is likely caused due to some topics containing fewer number of jobs in practice and choosing 250 of the most relevant jobs within a topic that should only have 50 jobs can result in jobs not related to the topic being included.

The figures 12, 13, and 14 in Appendix 3 show how the model proposed can incorporate the person's experience in hospitality and commerce and create tailored recommendations for each specific field. Her experience in tourism and aviation could be the reason why she is recommended for jobs in the "Bilingual" field. This ability to generate recommendations on a subset of the jobs implies that a student with a bachelor's degree in marketing that is working part-time in commerce and studying a master's degree in data science can receive accurate recommendations for each of the fields the student has experience in.

CONCLUSION

ANSWERS TO RESEARCH QUESTION

The results confirm that the hypothesis of adding topic modeling to a content-based recommender system for job recommendations outperforms a standard content-based recommender. This method has shown to produce more relevant recommendations due to the segmentation of recommendations into topics. These topics allow for recommendations to be less biased by unrelated or unhelpful terms which result in the model outperforming the baseline, particularly when an applicant has a varied background. The proposed model of content-based recommender system has applications in many fields where the content is highly variable, and the user requires a highly tailored output.

Future Work

The proposed recommender system is able to highlight important skills to have for a particular field. We did not incorporate this into the recommendations, but future studies can create more nuanced recommendations whereby the user is also suggested what to highlight in their application and what skills they would need to acquire to have a better chance of getting the job. It would also be possible to test different topic modelling techniques and metrics to calculate the similarities. Another possibility would be to test if the topic modeling techniques that performed worse in terms of generating relevant topics would be able to generate more relevant job recommendations compared to the best topic modelling method. However, it would remain difficult to determine the actual performance of how good the job recommendations are. That would make it difficult to also compare the performance of which topic model results in the better job recommendations.

Limitations

The result is limited by the nature of how difficult it is to evaluate the performance of recommender systems. The recommendations generated in this paper relied on a small dataset of jobs originating from a subset of locations in Europe. Launching the proposed recommender system into production on a global scale would require a significantly larger dataset. Only two CVs were used in this paper to generate recommendations. Before launching a system such as the one proposed in this paper, it is worth seeing how the system performs on other datasets. Testing the job recommendations with more CVs would give more insights into whether the model or data is working better within some categories of work.

CITATIONS

Abu Kausar, Mohammad & Dhaka, Vijaypal & Singh, Sanjeev. (2013). Web Crawler: A Review. *International Journal of Computer Applications.* 63. 31-36. 10.5120/10440-5125.

Aggarwal, C., 2016. Content-Based Recommender Systems. *Recommender Systems*, pp.139-166.

Al-Otaibi, S. T., & Ykhlef, M. (2012). A survey of job recommender systems. *International Journal of Physical Sciences*, *7*(29), 5127-5142.

Angelov, D. (2020). Top2vec: Distributed representations of topics. arXiv preprint arXiv:2008.09470.

Bag, S., Kumar, S. K., & Tiwari, M. K. (2019). An efficient recommendation generation using relevant Jaccard similarity. *Information Sciences*, *483*, 53-64.

Bansal, S., Srivastava, A., & Arora, A. (2017). Topic modeling driven content based jobs recommendation engine for recruitment industry. *Procedia computer science*, *122*, 865-872.

Blei, David & Ng, Andrew & Jordan, Michael & Lafferty, John. (2003). Journal of Machine Learning Research 3 (2003) 993-1022 Submitted 2/02; Published 1/03 Latent Dirichlet Allocation.

Ceri, S., Bozzon, A., Brambilla, M., Della Valle, E., Fraternali, P., Quarteroni, S. (2013). *Search Engines. In: Web Information Retrieval. Data-Centric Systems and Applications*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-39314-3_6.

Chang, J., Boyd_graber, J., Gerrish, S., Wang, C., & Blei, D. M. (2009). Reading Tea Leaves: How Humans Interpret Topic Models. In Neural Information Processing Systems (p. 9).

David M. Blei. 2012. Probabilistic topic models. Commun. ACM 55, 4 (April 2012), 77–84. https://doi.org/10.1145/2133806.2133826

Diaby, M., Viennet, E., & Launay, T. (2013, August). Toward the next generation of recruitment tools: an online social network-based job recommender system. In *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)* (pp. 821-828). IEEE.

Elliot Alderson, software developer - resumeviking.com. (n.d.). Retrieved May 29, 2022, from https://www.resumeviking.com/wp-content/uploads/2017/12/Elliot-Alderson-Resume-Software-Developer-2.pdf

Gillis, N., 2014. The why and how of nonnegative matrix factorization. *Regularization, optimization, kernels, and support vector machines*, *12*(257), pp.257-291.

Google. (n.d.). Background: What is a Generative Model? &nbsp;|&nbsp; Generative Adversarial Networks &nbsp;|&nbsp; Google Developers. Google. Retrieved May 27, 2022, from https://developers.google.com/machine-learning/gan/generative

Google Developers. 2018. *Collaborative Filtering Advantages & Disadvantages | Recommendation Systems | Google Developers*. [online] Available at: <https://developers.google.com/machine-learning/recommendation/collaborative/summary> [Accessed 28 May 2022].

Jurafsky, D., & Martin, J. H. (2021). Chapter 6. In *Speech and language processing* (3rd ed.). Essay, https://web.stanford.edu/~jurafsky/slp3/ .

Kuang, Da & Brantingham, P. & Bertozzi, Andrea. (2017). Crime Topic Modeling. Crime Science. 6. 10.1186/s40163-017-0074-0.

Le, Q., & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196). PMLR.

Lops, P., Jannach, D., Musto, C., Bogers, T. and Koolen, M., 2019. Trends in content-based recommendation. *User Modeling and User-Adapted Interaction*, 29(2), pp.239-249.

Manning, C. D., Raghavan, P., & Schutze, H. (2009). *An Introduction to Information Retrieval* (Online ed.). Cambridge University Press.

Mishra, R., & Rathi, S. (2020). Efficient and scalable job recommender system using collaborative filtering. In *ICDSMLA 2019* (pp. 842-856). Springer, Singapore.

Roeder, M., Both, A., & Hinneburg, A. (2015, February). Exploring the Space of Topic Coherence Measures. *ACM International WSDM Conference*. http://svn.aksw.org/papers/2015/WSDM_Topic_Evaluation/public.pdf

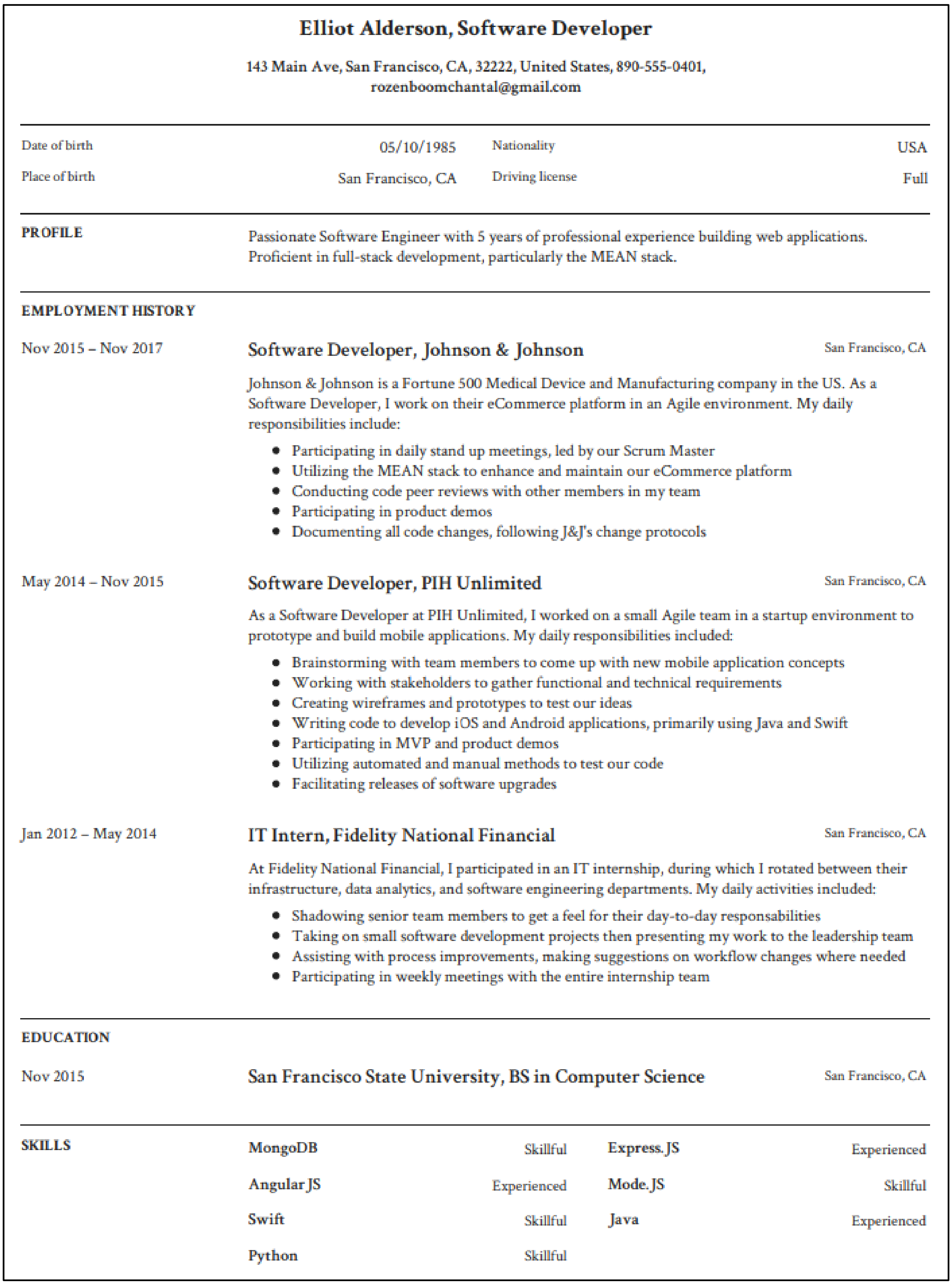
Sripada, S., Kasturi, V. G., & Parai, G. K. (2005). Multi-document extraction based Summarization. *CS 224N, Final Project*.

Türkmen, A. C. (2015). A review of nonnegative matrix factorization methods for clustering. arXiv preprint arXiv:1507.03194.

Appendices

Appendix 1. Sample CVs

Figure 1



Elliot Alderson, Software Developer

143 Main Ave, San Francisco, CA, 32222, United States, 890-555-0401,
rozenboomchantal@gmail.com

Date of birth	05/10/1985	Nationality	USA
Place of birth	San Francisco, CA	Driving license	Full

PROFILE

Passionate Software Engineer with 5 years of professional experience building web applications. Proficient in full-stack development, particularly the MEAN stack.

EMPLOYMENT HISTORY

Nov 2015 – Nov 2017

Software Developer, Johnson & Johnson San Francisco, CA

Johnson & Johnson is a Fortune 500 Medical Device and Manufacturing company in the US. As a Software Developer, I work on their eCommerce platform in an Agile environment. My daily responsibilities include:

- Participating in daily stand up meetings, led by our Scrum Master
- Utilizing the MEAN stack to enhance and maintain our eCommerce platform
- Conducting code peer reviews with other members in my team
- Participating in product demos
- Documenting all code changes, following J&J's change protocols

May 2014 – Nov 2015

Software Developer, PIH Unlimited San Francisco, CA

As a Software Developer at PIH Unlimited, I worked on a small Agile team in a startup environment to prototype and build mobile applications. My daily responsibilities included:

- Brainstorming with team members to come up with new mobile application concepts
- Working with stakeholders to gather functional and technical requirements
- Creating wireframes and prototypes to test our ideas
- Writing code to develop iOS and Android applications, primarily using Java and Swift
- Participating in MVP and product demos
- Utilizing automated and manual methods to test our code
- Facilitating releases of software upgrades

Jan 2012 – May 2014

IT Intern, Fidelity National Financial San Francisco, CA

At Fidelity National Financial, I participated in an IT internship, during which I rotated between their infrastructure, data analytics, and software engineering departments. My daily activities included:

- Shadowing senior team members to get a feel for their day-to-day responsabilities
- Taking on small software development projects then presenting my work to the leadership team
- Assisting with process improvements, making suggestions on workflow changes where needed
- Participating in weekly meetings with the entire internship team

EDUCATION

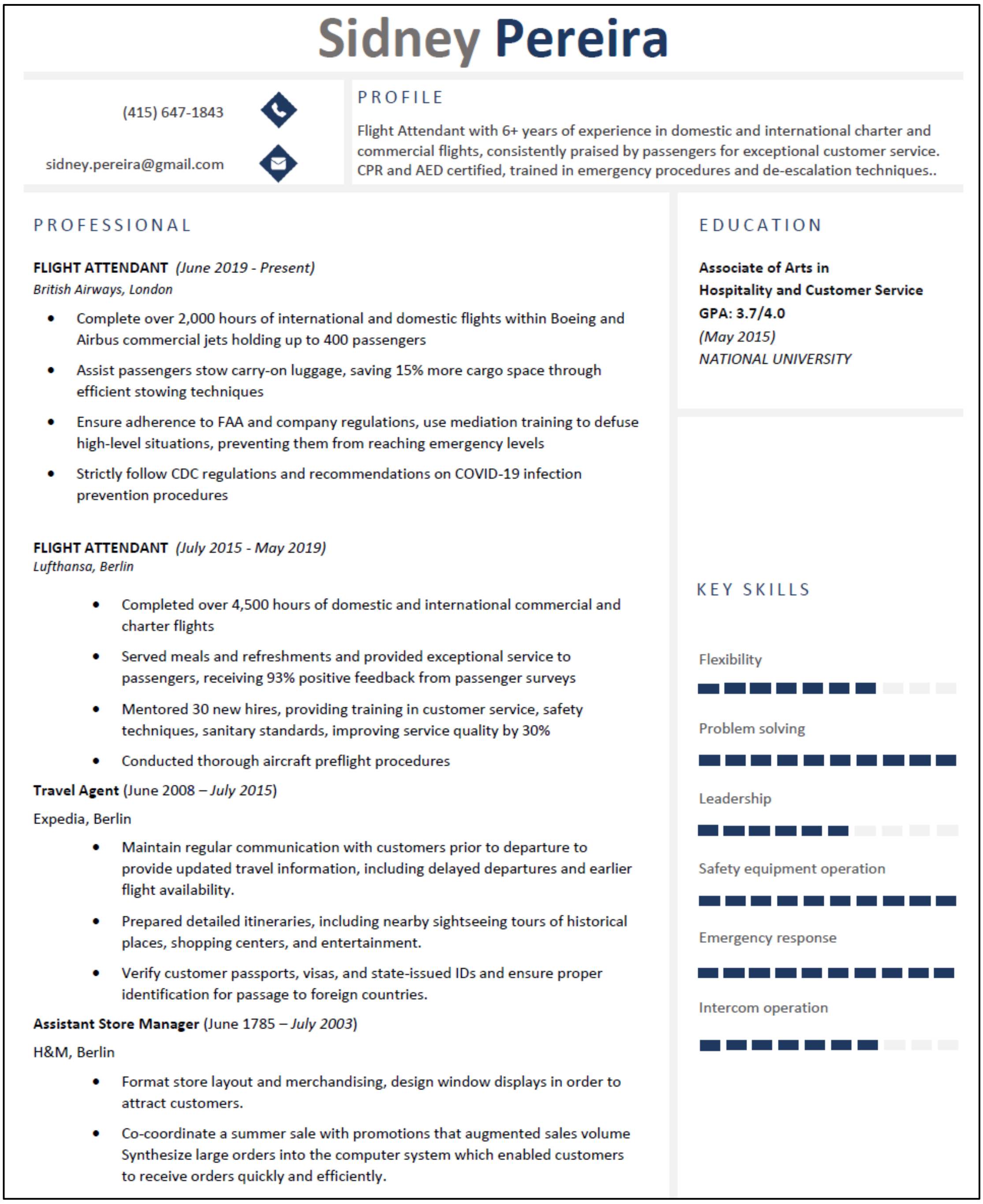
Nov 2015

San Francisco State University, BS in Computer Science San Francisco, CA

SKILLS

MongoDB	Skillful	Express.JS	Experienced
Angular JS	Experienced	Mode.JS	Skillful
Swift	Skillful	Java	Experienced
Python	Skillful		

Figure 2



Sidney Pereira

(415) 647-1843 ☎

sidney.pereira@gmail.com ✉

PROFILE

Flight Attendant with 6+ years of experience in domestic and international charter and commercial flights, consistently praised by passengers for exceptional customer service. CPR and AED certified, trained in emergency procedures and de-escalation techniques..

PROFESSIONAL

FLIGHT ATTENDANT *(June 2019 - Present)*
British Airways, London

- Complete over 2,000 hours of international and domestic flights within Boeing and Airbus commercial jets holding up to 400 passengers
- Assist passengers stow carry-on luggage, saving 15% more cargo space through efficient stowing techniques
- Ensure adherence to FAA and company regulations, use mediation training to defuse high-level situations, preventing them from reaching emergency levels
- Strictly follow CDC regulations and recommendations on COVID-19 infection prevention procedures

FLIGHT ATTENDANT *(July 2015 - May 2019)*
Lufthansa, Berlin

- Completed over 4,500 hours of domestic and international commercial and charter flights
- Served meals and refreshments and provided exceptional service to passengers, receiving 93% positive feedback from passenger surveys
- Mentored 30 new hires, providing training in customer service, safety techniques, sanitary standards, improving service quality by 30%
- Conducted thorough aircraft preflight procedures

Travel Agent (June 2008 – *July 2015*)

Expedia, Berlin

- Maintain regular communication with customers prior to departure to provide updated travel information, including delayed departures and earlier flight availability.
- Prepared detailed itineraries, including nearby sightseeing tours of historical places, shopping centers, and entertainment.
- Verify customer passports, visas, and state-issued IDs and ensure proper identification for passage to foreign countries.

Assistant Store Manager (June 1785 – *July 2003*)

H&M, Berlin

- Format store layout and merchandising, design window displays in order to attract customers.
- Co-coordinate a summer sale with promotions that augmented sales volume Synthesize large orders into the computer system which enabled customers to receive orders quickly and efficiently.

EDUCATION

Associate of Arts in
Hospitality and Customer Service
GPA: 3.7/4.0
(May 2015)
NATIONAL UNIVERSITY

KEY SKILLS

Flexibility
■■■■■■■□□□

Problem solving
■■■■■■■■■■

Leadership
■■■■■■□□□□

Safety equipment operation
■■■■■■■■■■

Emergency response
■■■■■■■■■■

Intercom operation
■■■■■■■□□□

Appendix 2. Topic modelling

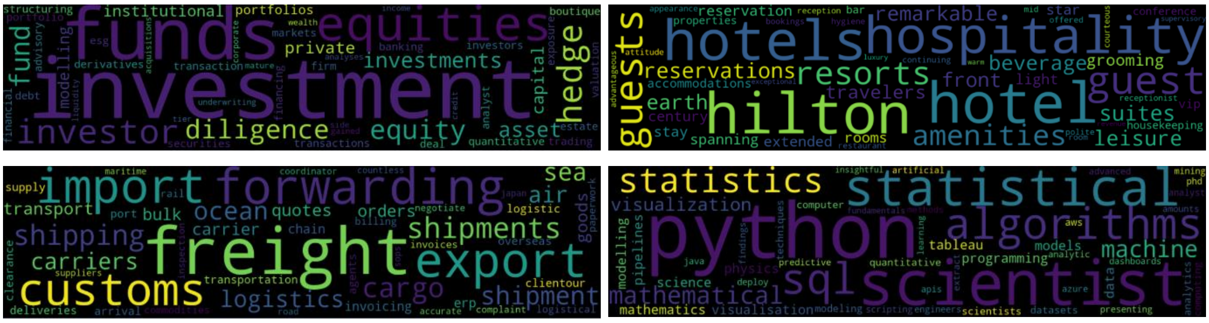


Figure 3. Sample of topics generated from the Top2Vec model (144 topics)

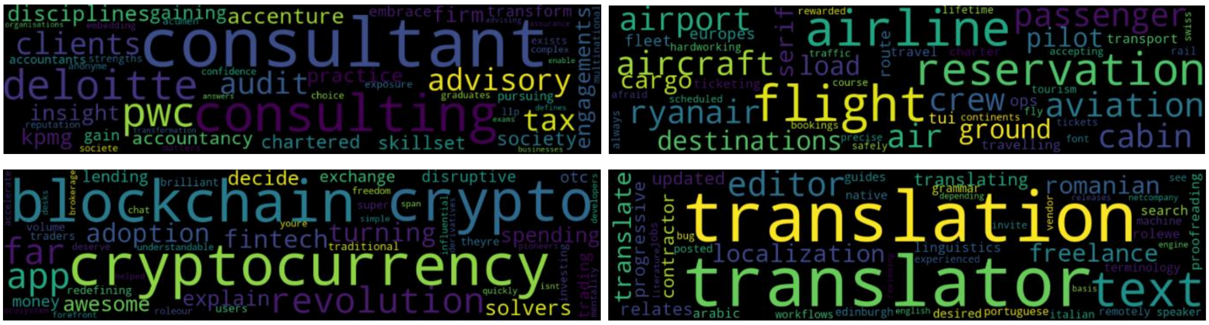


Figure 4. Sample of topics generated from the Top2Vec model (70 topics)

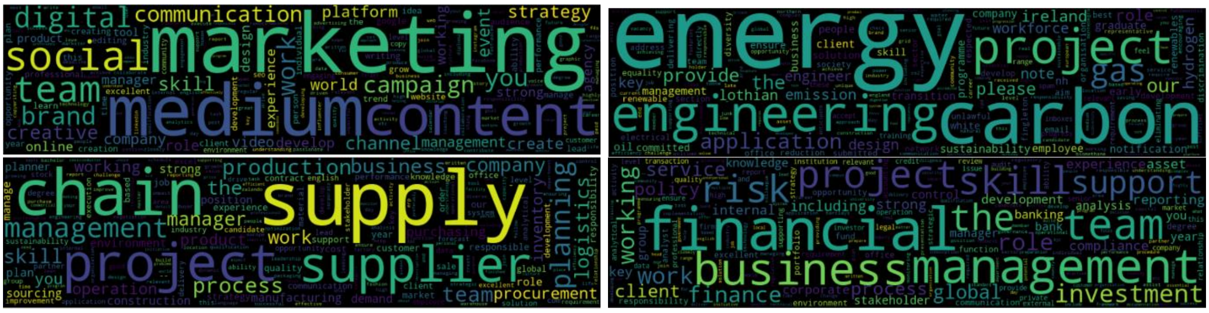


Figure 5. Sample of topics generated from the LDA model (70 topics)

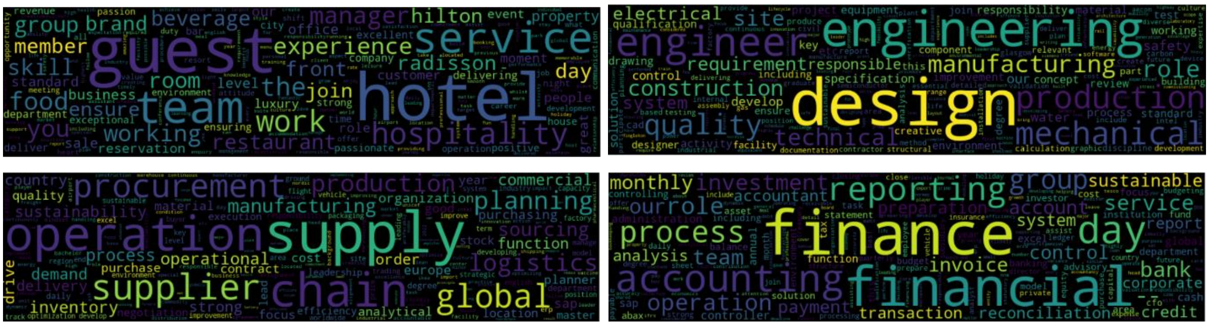
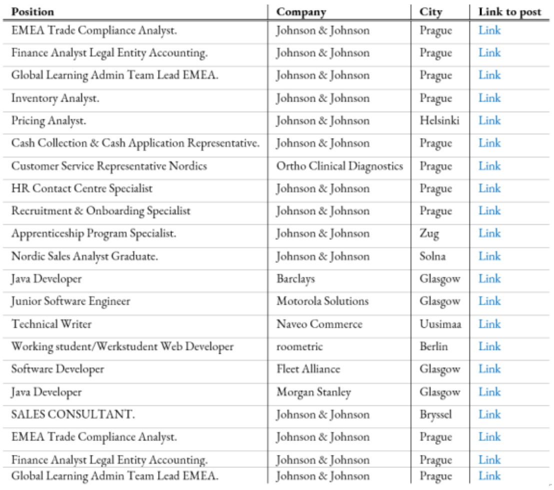


Figure 6. Sample of topics generated from the NMF model (70 topics)

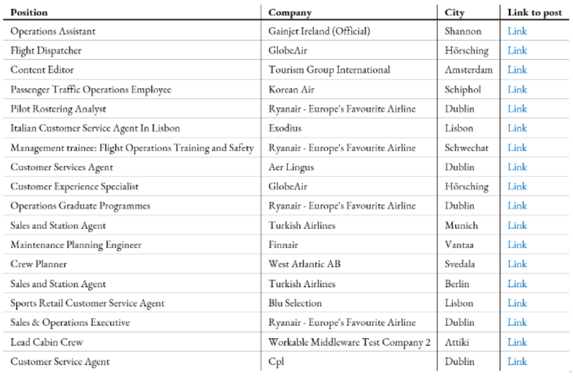
APPENDIX 3 RECOMMENDER SYSTEM

Base model



Position	Company	City	Link to post
EMEA Trade Compliance Analyst.	Johnson & Johnson	Prague	Link
Finance Analyst Legal Entity Accounting.	Johnson & Johnson	Prague	Link
Global Learning Admin Team Lead EMEA.	Johnson & Johnson	Prague	Link
Inventory Analyst.	Johnson & Johnson	Prague	Link
Pricing Analyst.	Johnson & Johnson	Helsinki	Link
Cash Collection & Cash Application Representative.	Johnson & Johnson	Prague	Link
Customer Service Representative Nordics	Ortho Clinical Diagnostics	Prague	Link
HR Contact Centre Specialist	Johnson & Johnson	Prague	Link
Recruitment & Onboarding Specialist	Johnson & Johnson	Prague	Link
Apprenticeship Program Specialist.	Johnson & Johnson	Zug	Link
Nordic Sales Analyst Graduate.	Johnson & Johnson	Solna	Link
Java Developer	Barclays	Glasgow	Link
Junior Software Engineer	Motorola Solutions	Glasgow	Link
Technical Writer	Naveo Commerce	Uusimaa	Link
Working student/Werkstudent Web Developer	roometric	Berlin	Link
Software Developer	Fleet Alliance	Glasgow	Link
Java Developer	Morgan Stanley	Glasgow	Link
SALES CONSULTANT.	Johnson & Johnson	Bryssel	Link
EMEA Trade Compliance Analyst.	Johnson & Johnson	Prague	Link
Finance Analyst Legal Entity Accounting.	Johnson & Johnson	Prague	Link
Global Learning Admin Team Lead EMEA.	Johnson & Johnson	Prague	Link

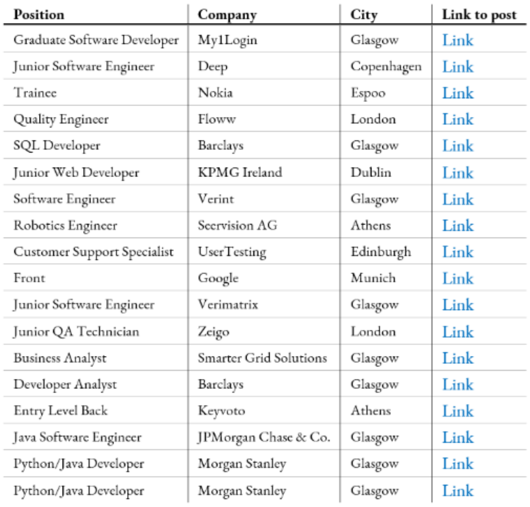
Figure 7 Recommendations by the base model for CV 1



Position	Company	City	Link to post
Operations Assistant	Gainjet Ireland (Official)	Shannon	Link
Flight Dispatcher	GlobeAir	Hörsching	Link
Content Editor	Tourism Group International	Amsterdam	Link
Passenger Traffic Operations Employee	Korean Air	Schiphol	Link
Pilot Rostering Analyst	Ryanair - Europe's Favourite Airline	Dublin	Link
Italian Customer Service Agent In Lisbon	Exodius	Lisbon	Link
Management trainee: Flight Operations Training and Safety	Ryanair - Europe's Favourite Airline	Schwechat	Link
Customer Services Agent	Aer Lingus	Dublin	Link
Customer Experience Specialist	GlobeAir	Hörsching	Link
Operations Graduate Programmes	Ryanair - Europe's Favourite Airline	Dublin	Link
Sales and Station Agent	Turkish Airlines	Munich	Link
Maintenance Planning Engineer	Finnair	Vantaa	Link
Crew Planner	West Atlantic AB	Svedala	Link
Sales and Station Agent	Turkish Airlines	Berlin	Link
Sports Retail Customer Service Agent	Blu Selection	Lisbon	Link
Sales & Operations Executive	Ryanair - Europe's Favourite Airline	Dublin	Link
Lead Cabin Crew	Workable Middleware Test Company 2	Attiki	Link
Customer Service Agent	Cpl	Dublin	Link

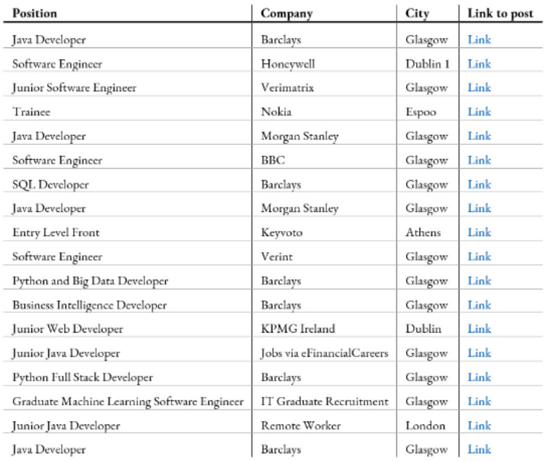
Figure 8 Recommendations by the base model for CV 2

Proposed model using topic modelling



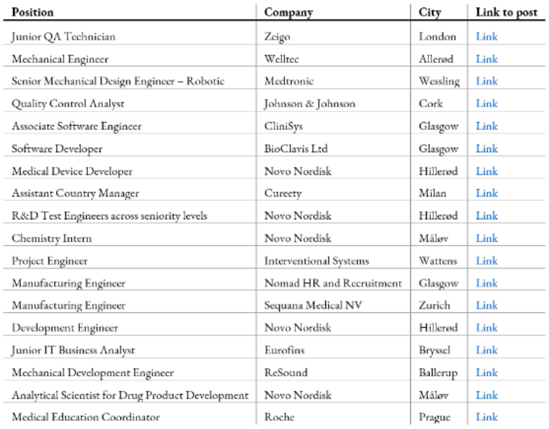
Position	Company	City	Link to post
Graduate Software Developer	My1Login	Glasgow	Link
Junior Software Engineer	Deep	Copenhagen	Link
Trainee	Nokia	Espoo	Link
Quality Engineer	Floww	London	Link
SQL Developer	Barclays	Glasgow	Link
Junior Web Developer	KPMG Ireland	Dublin	Link
Software Engineer	Verint	Glasgow	Link
Robotics Engineer	Seervision AG	Athens	Link
Customer Support Specialist	UserTesting	Edinburgh	Link
Front	Google	Munich	Link
Junior Software Engineer	Verimatrix	Glasgow	Link
Junior QA Technician	Zeigo	London	Link
Business Analyst	Smarter Grid Solutions	Glasgow	Link
Developer Analyst	Barclays	Glasgow	Link
Entry Level Back	Keyvoto	Athens	Link
Java Software Engineer	JPMorgan Chase & Co.	Glasgow	Link
Python/Java Developer	Morgan Stanley	Glasgow	Link
Python/Java Developer	Morgan Stanley	Glasgow	Link

Figure 9 Recommendations by the topic-based model for CV 1 (topic 1: "IT Services)



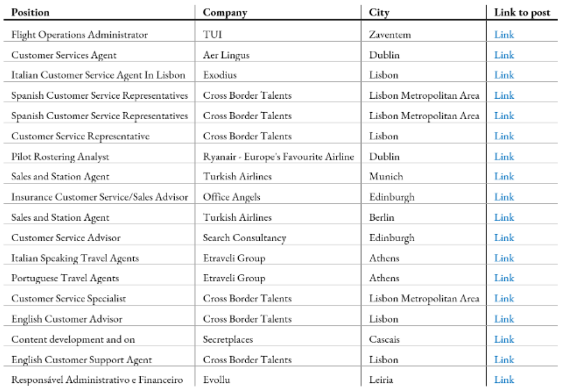
Position	Company	City	Link to post
Java Developer	Barclays	Glasgow	Link
Software Engineer	Honeywell	Dublin 1	Link
Junior Software Engineer	Verimatrix	Glasgow	Link
Trainee	Nokia	Espoo	Link
Java Developer	Morgan Stanley	Glasgow	Link
Software Engineer	BBC	Glasgow	Link
SQL Developer	Barclays	Glasgow	Link
Java Developer	Morgan Stanley	Glasgow	Link
Entry Level Front	Keyvoto	Athens	Link
Software Engineer	Verint	Glasgow	Link
Python and Big Data Developer	Barclays	Glasgow	Link
Business Intelligence Developer	Barclays	Glasgow	Link
Junior Web Developer	KPMG Ireland	Dublin	Link
Junior Java Developer	Jobs via eFinancialCareers	Glasgow	Link
Python Full Stack Developer	Barclays	Glasgow	Link
Graduate Machine Learning Software Engineer	IT Graduate Recruitment	Glasgow	Link
Junior Java Developer	Remote Worker	London	Link
Java Developer	Barclays	Glasgow	Link

Figure 10 Recommendations by the topic-based model for CV 1 (topic 2: "Data Science")



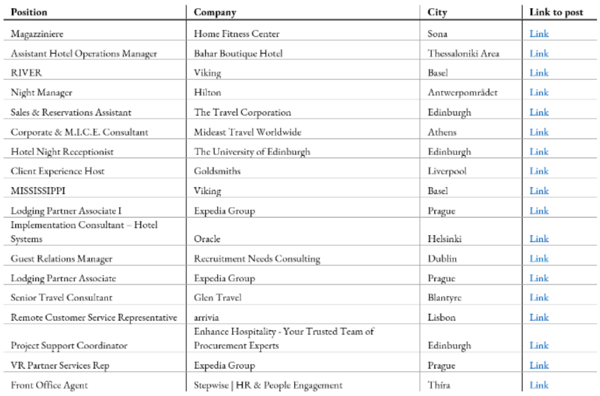
Position	Company	City	Link to post
Junior QA Technician	Zeigo	London	Link
Mechanical Engineer	Welltec	Allerød	Link
Senior Mechanical Design Engineer – Robotic	Medtronic	Wessling	Link
Quality Control Analyst	Johnson & Johnson	Cork	Link
Associate Software Engineer	CliniSys	Glasgow	Link
Software Developer	BioClavis Ltd	Glasgow	Link
Medical Device Developer	Novo Nordisk	Hillerød	Link
Assistant Country Manager	Cureety	Milan	Link
R&D Test Engineers across seniority levels	Novo Nordisk	Hillerød	Link
Chemistry Intern	Novo Nordisk	Måløv	Link
Project Engineer	Interventional Systems	Wattens	Link
Manufacturing Engineer	Nomad HR and Recruitment	Glasgow	Link
Manufacturing Engineer	Sequana Medical NV	Zurich	Link
Development Engineer	Novo Nordisk	Hillerød	Link
Junior IT Business Analyst	Eurofins	Bryssel	Link
Mechanical Development Engineer	ReSound	Ballerup	Link
Analytical Scientist for Drug Product Development	Novo Nordisk	Måløv	Link
Medical Education Coordinator	Roche	Prague	Link

Figure 11 Recommendations by the topic-based model for CV 1 (topic 3: "Logistics")



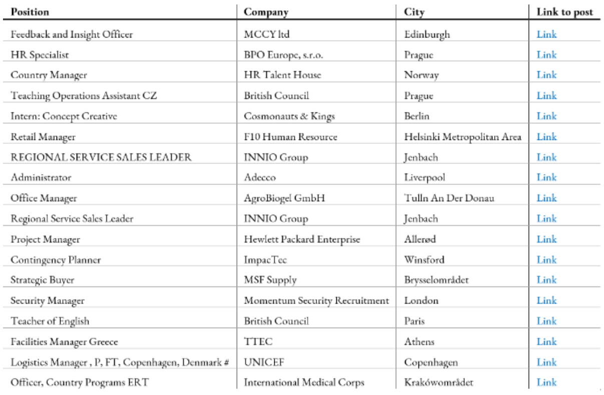
Position	Company	City	Link to post
Flight Operations Administrator	TUI	Zaventem	Link
Customer Services Agent	Aer Lingus	Dublin	Link
Italian Customer Service Agent In Lisbon	Exodius	Lisbon	Link
Spanish Customer Service Representatives	Cross Border Talents	Lisbon Metropolitan Area	Link
Spanish Customer Service Representatives	Cross Border Talents	Lisbon Metropolitan Area	Link
Customer Service Representative	Cross Border Talents	Lisbon	Link
Pilot Rostering Analyst	Ryanair - Europe's Favourite Airline	Dublin	Link
Sales and Station Agent	Turkish Airlines	Munich	Link
Insurance Customer Service/Sales Advisor	Office Angels	Edinburgh	Link
Sales and Station Agent	Turkish Airlines	Berlin	Link
Customer Service Advisor	Search Consultancy	Edinburgh	Link
Italian Speaking Travel Agents	Etraveli Group	Athens	Link
Portuguese Travel Agents	Etraveli Group	Athens	Link
Customer Service Specialist	Cross Border Talents	Lisbon Metropolitan Area	Link
English Customer Advisor	Cross Border Talents	Lisbon	Link
Content development and on	Secretplaces	Cascais	Link
English Customer Support Agent	Cross Border Talents	Lisbon	Link
Responsável Administrativo e Financeiro	Evollu	Leiria	Link

Figure 12 Recommendations by the topic-based model for CV 2 (topic 1: "Bilingual")



Position	Company	City	Link to post
Magazziniere	Home Fitness Center	Sona	Link
Assistant Hotel Operations Manager	Bahar Boutique Hotel	Thessaloniki Area	Link
RIVER	Viking	Basel	Link
Night Manager	Hilton	Antwerpområdet	Link
Sales & Reservations Assistant	The Travel Corporation	Edinburgh	Link
Corporate & M.I.C.E. Consultant	Mideast Travel Worldwide	Athens	Link
Hotel Night Receptionist	The University of Edinburgh	Edinburgh	Link
Client Experience Host	Goldsmiths	Liverpool	Link
MISSISSIPPI	Viking	Basel	Link
Lodging Partner Associate I	Expedia Group	Prague	Link
Implementation Consultant – Hotel Systems	Oracle	Helsinki	Link
Guest Relations Manager	Recruitment Needs Consulting	Dublin	Link
Lodging Partner Associate	Expedia Group	Prague	Link
Senior Travel Consultant	Glen Travel	Blantyre	Link
Remote Customer Service Representative	arrivia	Lisbon	Link
Project Support Coordinator	Enhance Hospitality - Your Trusted Team of Procurement Experts	Edinburgh	Link
VR Partner Services Rep	Expedia Group	Prague	Link
Front Office Agent	Stepwise | HR & People Engagement	Thíra	Link

Figure 13 Recommendations by the topic-based model for CV 2 (topic 2: "Hospitality")



Position	Company	City	Link to post
Feedback and Insight Officer	MCCY ltd	Edinburgh	Link
HR Specialist	BPO Europe, s.r.o.	Prague	Link
Country Manager	HR Talent House	Norway	Link
Teaching Operations Assistant CZ	British Council	Prague	Link
Intern: Concept Creative	Cosmonauts & Kings	Berlin	Link
Retail Manager	F10 Human Resource	Helsinki Metropolitan Area	Link
REGIONAL SERVICE SALES LEADER	INNIO Group	Jenbach	Link
Administrator	Adecco	Liverpool	Link
Office Manager	AgroBiogel GmbH	Tulln An Der Donau	Link
Regional Service Sales Leader	INNIO Group	Jenbach	Link
Project Manager	Hewlett Packard Enterprise	Allerød	Link
Contingency Planner	ImpacTec	Winsford	Link
Strategic Buyer	MSF Supply	Brysselområdet	Link
Security Manager	Momentum Security Recruitment	London	Link
Teacher of English	British Council	Paris	Link
Facilities Manager Greece	TTEC	Athens	Link
Logistics Manager , P, FT, Copenhagen, Denmark #	UNICEF	Copenhagen	Link
Officer, Country Programs ERT	International Medical Corps	Krakówområdet	Link

Figure 14 Recommendations by the topic-based model for CV 2 (topic 3: "Commerce")