Knowledge Discovery from CVs: A Topic Modeling Procedure

Alexander Schiller

University of Regensburg, Department of Management Information Systems,
Regensburg, Germany
alexander.schiller@wiwi.uni-regensburg.de

Abstract. With a huge number of CVs available online, recruiting via the web has become an integral part of human resource management for companies. Automated text mining methods can be used to analyze large databases containing CVs. We present a topic modeling procedure consisting of five steps with the aim of identifying competences in CVs in an automated manner. Both the procedure and its exemplary application to CVs from IT experts are described in detail. The specific characteristics of CVs are considered in each step for optimal results. The exemplary application suggests that clearly interpretable topics describing fine-grained competences (e.g., Java programming, web design) can be discovered. This information can be used to rapidly assess the contents of a CV, categorize CVs and identify candidates for job offers. Furthermore, a topic-based search technique is evaluated to provide helpful decision support.

Keywords: Text Mining, Topic Modeling, Latent Dirichlet Allocation, Human Resource Management

1 Introduction

Acquiring the right personnel is one of the most critical success factors for companies [1, 2]. In this area, recruiting via the web has gained significant importance over the last years [3]. It is not only of practical interest, but has also received much scientific attention (cf., e.g., [4, 5]). Opportunities are, for example, provided by well-known professional online social networks such as *LinkedIn* and *XING*, which are becoming highly popular. For instance, as of Q2 2018, *LinkedIn* has more than 562 million members [6]. Recruiting via the web is further made possible by CVs provided not only on these networks, but also on private homepages and websites specializing in making available a wide range of CVs (e.g., *Indeed*, *CareerBuilder*, *Monster*). Overall, a huge number of CVs can be acquired online. Based on these CVs, companies have the prospect to identify promising job candidates and to conduct proactive recruiting.

However, to capitalize on this potential, a very large amount of semi- and unstructured data needs to be analyzed. While approaches for a manual analysis of document collections exist [7], this task becomes too time-consuming for large collections of complex documents (such as CVs) [8]. This issue is addressed by

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

automated text mining methods, which have already been used successfully for human resource management (HRM) [9, 10]. In particular, topic modeling approaches such as Latent Dirichlet Allocation (LDA) are promising in this application context. They are able to discover the hidden thematic structure present in a document collection [11]. Thus, they should be able to extract key information from CVs. More precisely, high-quality, fine-grained topics in a specialized topic model should represent skills, abilities, knowledge and work expertise (in the following subsumed by "competences" as in [12]). This information can then be used to, for instance, rapidly assess the contents of a CV, categorize CVs and identify candidates for job offers. Topic models offer unique advantages compared to a keyword search on existing platforms. However, research has not yet discussed the application of topic modeling to CVs, leaving open crucial issues (cf. Section 2.2). This paper thus focuses on the following research question: *How can topic modeling be used to discover knowledge from CVs?*

The remainder of the paper is structured as follows. In the next section, we discuss the problem context as well as related work and the research gap. In Section 3, we propose a procedure for knowledge discovery from CVs. Section 4 contains an application of the procedure and an evaluation of the results. Finally, we provide conclusions, limitations and directions for future research.

2 Background

In this section, we first briefly introduce topic modeling, LDA and evaluation methods for topic models. Then, we give an overview of related literature and discuss the research gap.

2.1 Problem Context

Topic modeling approaches aim to discover the latent thematic structure in a document collection and to identify thematically similar documents [11]. A topic model consists of a number of topics, each represented by terms strongly associated to the topic. Recently, probabilistic approaches have been highly popular. Here, topics can be seen as probability distributions over terms and documents as probability distributions over topics. In this paper, we focus on LDA [13], a probabilistic approach not relying on any kind of training data. It is the most widely-applied topic modeling approach [14]. Distributions are calculated using sampling or optimization procedures which take into account term-document-frequencies [15]. LDA is based on the bag-of-words model (i.e., the order of words in documents is ignored). This makes it particularly suitable for CVs, which often are formulated in note form instead of continuous text.

While an evaluation of topic models by humans is considered the gold standard [16], the required time effort has sparked the need for automated evaluations, especially for testing a large number of pre-processing and parameter configurations. Many criteria and methods have been proposed, discussing, for instance, the similarity [17], stability [14] or semantic coherence [18–20] of topics. While a negative correlation to human interpretability has been reported for some methods [16], semantic coherence has been

shown to provide assessments of high quality [8, 19–21]. It is often calculated based on normalized pointwise mutual information (NPMI) [19, 21]. The idea is that a topic is of higher quality when the terms strongly associated to the topic often occur in close proximity in a text corpus. Following the discussion above, measuring semantic coherence by NPMI is used in this paper for assessing various topic model configurations before the final topic model is humanly interpreted.

2.2 Related Work and Research Gap

Topic modeling and in particular LDA is widely applicable to a large range of contexts and has been successfully employed to, for instance, consumer good reviews [8], research articles [22], hotel critiques [23] and even in BPM [24]. A tutorial for applying LDA in IS in general has been proposed as well [8]. However, the high degree of abstraction and flexibility also come at a price: An adaption to the application context is necessary to provide proper results. Despite much existing work to use text mining in HRM [9, 10], a literature search revealed that little of it has addressed topic modeling.

A notable exception is a work suggesting topic modeling for job offers [4]. The objective was to identify competences of importance in the construction industry. This research is similar to ours in the sense that a topic modeling approach was used for this task. However, the aim differs, because instead of CVs, job offers have been analyzed. Furthermore, no procedure for knowledge discovery is described.

In further research, *LinkedIn* profiles of BPM professionals are examined via topic modeling to investigate the role of gender in BPM [12]. While here, a topic modeling approach is applied to documents which are similar to CVs, the aim of the research is completely different to ours. Usability for recruiting is not discussed and no procedure for knowledge discovery is presented.

To sum up, none of the existing works has proposed a procedure for knowledge discovery from CVs using topic modeling. Thus, following existing literature in this regard leads to multiple crucial issues, which constitutes the research gap addressed by this paper: First, not considering the type and characteristics of documents to be analyzed produces non-optimal results. For instance, this is due to generic text pre-processing (e.g., in the case of CVs, no removal of author's contact details). Second, existing approaches cannot even be readily applied, as essential steps are not described. For example, the acquisition of CVs and their general pre-processing is not discussed in existing topic modeling literature. Finally, the goals of applying topic modeling to CVs are not considered in existing works. This means that, critically, it remains unclear how to use the topic modeling results for actual benefit in HRM (e.g., for recruiting).

3 Knowledge Discovery from CVs

Subsequently, our procedure for knowledge discovery from CVs is presented. Figure 1 illustrates the five steps of the procedure, adapted from the process for topic modeling in general IS as proposed in [8]. After the initial acquisition of CVs (i), general pre-processing (ii) and text pre-processing (iii) are required. Only then, the pre-

processed CVs can be analyzed by applying LDA (iv). Finally, the results of the application can be interpreted and used (v). The five steps are described in detail in the following.

Figure 1. Steps for knowledge discovery from CVs

3.1 Acquisition of CVs

With unstructured data proliferating on the web, many options for acquiring CVs are available. The most prominent ones are utilizing (a) professional social networks such as *LinkedIn*, (b) specialized portals such as *Indeed* and (c) a web crawler.

(a): Professional social networks offer members the opportunity to present themselves to companies and recruiters via disclosing information on their profile. This includes, in particular, past work experience, education, skills, abilities, publications and interests as well as contact information. The information can be extracted from the profiles to generate CV-like documents. Internal and external tools for extraction are readily available for the most common professional social networks (e.g., *LinkedIn*, *XING*). For example, *LinkedIn* itself offers a native functionality to export member profiles as CVs in PDF format. Moreover, these portals allow members to directly upload their CVs, which can then be accessed and stored.

(b): Many job portals (e.g., *Indeed*, *CareerBuilder*, *Monster*) encourage their users to post their CV. These portals offer a (keyword) search engine which can be used to obtain CVs. For example, in Q2 2018, an exemplary search for CVs with "Data Scientist" as job title in New York City produced over 1,100 hits on *Indeed*. CVs resulting from a search can be accessed and, subsequently, stored in PDF format.

(c): Another opportunity for the acquisition of publicly available CVs is the use of a web crawler. A web crawler is an automated program able to navigate the web and store relevant information. Specifically, such a web crawler can be fed with desired search terms and programmed to find and store PDFs including these search terms. This allows the acquisition of CVs from the general searchable web. In particular, also CVs available on private homepages can be found and stored.

3.2 General Pre-processing

Once a sufficient quantity of PDFs containing CVs has been obtained, a general pre-processing of this collection of data is required to ready the collection for text pre-processing and further analyses. Depending on the way the PDFs were obtained, the common challenges (C1) or (C2) may need to be resolved:

(C1) The language of the documents does not match, causing problems for many text pre-processing routines. To address this issue, an approach for automatic language identification [25, 26] can be used. A high quality of automatic language identification can be achieved as CVs contain a substantial number of words.

(C2) The collection of data does not exclusively consist of CVs (e.g., when the PDFs were obtained using a web crawler). Obviously, this issue can lead to unsatisfactory results in the later stages of analysis, for instance, when a job offer instead of a CV is erroneously assessed to be the optimal match for a search query. A human is able to almost instantly decide whether a PDF is a CV or a non-CV with a very high degree of confidence. However, a manual distinction of CVs and non-CVs may still not be promising due to the substantial time effort required for assessing a large collection of data by hand. Thus, automated methods such as classification algorithms can be used.

In any case, PDFs should be converted to a more manageable format (such as TXT) for further analyses, and be fitted in a database for storage. As suggested in [8], an exploratory data analysis may be performed to detect possible data quality issues and to obtain a general understanding of the data to be analyzed.

3.3 Text Pre-processing

Meaningful text pre-processing before the application of a topic modeling approach is of high importance [8]. This is particularly the case for CVs, which contain many terms or even whole components irrelevant to the envisioned goal of knowledge discovery. There are simple and well-known pre-processing routines which are accepted to be valuable for (almost) all kinds of texts. Common examples are the removal of formatting tags and special characters, tokenization (i.e., splitting up documents into words), lowercasing and the removal of words occurring only in few documents. These routines can be looked up in renowned sources [27, 28]; in the following, we discuss routines which possess special characteristics with regard to CVs more explicitly.

N-gram-Creation. N-grams (expressions consisting of two or more words) instead of single words can be considered for further text analysis. For instance, many competence descriptions (e.g., ability in software such as Visual Studio) in CVs contain multiple words. Thus, a thorough creation of n-grams may be of importance. However, care needs to be taken because many skill descriptions in CVs are often used together but are not a real expression. For example, disclosing language skills in English and Spanish is common, which might lead to an incorrect 2-gram "English Spanish".

Stop Word Removal. Words that occur frequently, but are uninformative and decrease the quality and interpretability of topics need to be removed. To achieve this goal in the context of CVs, multiple types of words have to be eliminated. The first type includes general language-specific stop words, which usually are words that have only a grammatical or syntactical function such as propositions. The second type are CV-specific stop words, which are words commonly occurring in all kinds of CVs (e.g., "resume", "name"). The third type are stop words specific for the CV database at hand. To identify these stop words, word frequency lists can be used [27]. Finally, numbers may or may not also be seen as "stop words". When competences are to be modeled as topics, numbers tend to obscure the results; thus, they should also be filtered.

Part-of-speech Filtering. Research has provided varying results with respect to which parts of speech should be filtered using LDA [8, 29]. Against this background and taking into account that CVs contain a word distribution different from other types of documents (e.g., higher prevalence of nouns), part-of-speech filtering needs to be

analyzed and adapted to obtain optimal results. In any case, nouns are not to be filtered as they transport essential information regarding competences.

Stemming & Lemmatization. Both stemming and lemmatization aim to decrease the number of considered terms by consolidating similar words. Stemming strives to truncate words to their stem, while lemmatization seeks to reduce words to their dictionary form. Stemming is seen as problematic for the application of LDA [27, 30, 31], for instance due to the danger that words with substantially different meaning are consolidated. Lemmatization, on the other hand, has mostly shown positive effects [29, 31]. However, CVs are structured differently than other types of documents, for example with respect to parts of speech; hence, the use of lemmatization should also be analyzed and adapted with respect to the database at hand to obtain optimal results.

Named Entity Recognition. Approaches for named entity recognition classify named entities in text into pre-defined categories (e.g., person names). The appearance of names in CVs is particular. CVs contain the name of the CV's author, possibly other person names (e.g., co-authors of publications), location names (referring to, e.g., company sites) and organization names. Location names may be useful in order to pinpoint expertise in certain areas such as the D-A-CH region. Organization names are of high relevance for many descriptions of competences (e.g., Microsoft Office). Person names, however, do not contribute to interpretable topics and should be filtered.

Overall, it has to be stated that – as it is usually the case in text mining – finding a very good pre-processing configuration for topic modeling of CVs is a non-trivial task. One has to experiment with different configurations to obtain optimal results with respect to the database at hand. Based on the discussion above, in the context of CVs, it seems particularly sensible to fix most steps but to experiment with n-gram creation, part-of-speech filtering and lemmatization.

3.4 Application of LDA

LDA requires as input two hyperparameters α and β as well as the total number of topics N. The shape of the CV-topic-distributions is determined by α [13]. When α is large, CVs are described by many topics and thus competences, whereas a small α leads to few topics per CV. Obviously, α should neither be too large (resulting in an unwieldy description of CVs which does not carve out the main competences) nor too small (resulting in only the most prominent competence being identified). The shape of the word-topic-distributions is controlled by β [13]. A large β implies that topics are widespread (i.e., competences are described broadly). A small β, in turn, leads to narrow topics and competences. In practice, α and β are often set to standard values (e.g., $1/N$) which have been shown to work well for a large range of application contexts [8]. Alternatively, an optimization can be performed [32].

If the number of topics N is too small, resulting topics may be general and widespread, representing a large variety of competences. For example, in a database containing CVs from IT experts, programming skills may constitute a single topic and not be differentiated further. As a result, topic distributions of CVs are not very meaningful: The competences of two persons portrayed by CVs with a similar topic distribution may still differ substantially. On the other hand, the more topics, the more

challenging it is for humans to grasp all word-topic-distributions and to interpret CV-topic-distributions. Moreover, if *N* is too large, resulting topics may be very similar to each other. Thus, (almost) the same competences can be represented by multiple topics, leading to interpretation difficulties. The competences of two persons portrayed by CVs with a largely differing topic distribution may still be similar. To determine a favorable number of topics *N*, evaluation methods for topic models (cf. Section 2.1) can be used. Then, the resulting topics can be analyzed with regard to their human interpretability. In particular, it can be checked whether competences of interest are represented by topics or pre-processing configuration and LDA application need to be refined.

3.5 Interpretation & Use of Results

Once the topic model has been constructed, the actual knowledge discovery can begin. The topic model provides both word-topic-distributions and CV-topic-distributions.

The word-topic-distributions can be analyzed to obtain an understanding of the subjects generally present in the CVs. On a more fine-grained level, analyses of each topic – in particular, of the words with the highest probability in each topic – can be conducted to allow for their interpretation. Ideally, many of the topics clearly represent specific competences (e.g., web development). It is to be expected that also other topics representing, for instance, university or school career are contained in the model. In any case, as long as topics are interpretable, they should be labelled accordingly. Preferably, multiple persons label topics independently and compare their assessments afterwards.

The CV-topic-distributions offer a succinct description of each CV's topics. They allow to analyze CVs with respect to contained topics, in particular with respect to the competences of the portrayed person. This is especially helpful when topics have already been labelled. Then, the competences of a person portrayed by a CV can be assessed rapidly by observing the respective CV-topic-distribution and taking into account the labels associated to the most prevalent topics. Such an assessment is useful in HRM (e.g., for swift decision support in regard to the relevance of applicants for a job offer). Here, it is also important to note that using LDA, the topic distribution of a fresh CV can be determined quickly without re-running the whole model. Moreover, based on CV-topic-distributions, CVs may be categorized or tagged for future use. For example, all CVs which exhibit a proportion above 40% for a topic representing web development skills can be marked as relevant for future job offers in this area.

Furthermore, based on word-topic-distributions and CV-topic-distributions, techniques for post-processing LDA results can be applied. This includes in particular visualization approaches (e.g., [33, 34]) which assist analysts in gaining an overview and interpreting. For instance, LDAvis [34] provides clear illustrations of word-topic-distributions and offers to display terms particularly characteristic for a topic based on a relevance metric. This can be helpful to pinpoint rare but valuable competences occurring almost exclusively in a certain topic. If a specialist possessing this rare skill is required, the respective CV can then be retrieved quickly.

Obviously, there are further possibilities yet to be explored. We describe the following technique of a *topic-based search for CVs* as an exemplary idea. To facilitate this technique, in a first step, topics of interest and interpretable as competences are

extracted from the LDA model and labelled. Based upon these topics, a query can be formulated which represents the desired competences in form of a search vector. For instance, if a Java developer with complementary competences in web development and web design is in demand, the emphasis may be 50% on Java programming, 30% on web development and 20% on web design. The search vector would thus include the values 0.5, 0.3 and 0.2 for topics representing Java programming, web development and web design respectively and 0 for all other topics. Then, the similarity between the search vector and the CV-topic-distributions of each CV can be calculated based on established similarity measures such as cosine similarity or Kullback-Leibler divergence for topics [17]. The most similar CVs can be manually screened and promising candidates can be contacted for recruiting. Such a topic-based search possesses clear advantages compared to a usual keyword search (e.g., also on platforms such as *LinkedIn* and *Indeed*), which can be illustrated by the example above:

1) The topic-based search allows to search for actual competences and not just for words which may or may not represent these competences reasonably well. For instance, a CV may not contain the term "web development" but the portrayed person may still report experience in JavaScript, HTML and PHP. The respective CV would be deemed irrelevant by a keyword search, whereas the topic-based search acknowledges the competence in web development.

2) In the topic-based search, it is possible to put emphasis on an aspect and the mere occurrence of a keyword is not enough for a CV to be relevant. For example, the specification that Java skills should make up for 50% of a CV's topic distribution means that CVs indeed need to contain a lot of Java-related terms to be assessed as relevant in the topic-based search. In contrast, a keyword search for terms such as Java is not very promising because a large number of CVs will claim at least some competence in Java.

3) The topic-based search allows to specify a weighting between different competences. In the example, Java programming is weighted with 50%, web development with 30% and web design with 20%. However, in a simple keyword search, each keyword would be treated as equally important. Weighting allows to more accurately identify the candidates which fit the requirements best (e.g., that Java development skills are most important and the other competences are complementary).

4 Application and Evaluation

In this section, we first describe how we exemplarily applied the procedure presented in Sections 3.1-3.5. In addition to this demonstration of practical applicability, we also evaluate the feasibility of the results within Section 4.5.

4.1 Acquisition of CVs

For our exemplary application, we decided to focus on CVs from IT experts for the following reasons. First, IT experts often possess diverse competences (e.g., programming languages, software, ...) which they report in their CVs. A topic modeling approach adapted to CVs should be able to identify these competences and categorize

them into interpretable topics. Second, focusing on a single area provides a particular challenge for the procedure. CVs portraying persons with completely different competences are relatively easy to distinguish. However, a procedure that provides good results even when applied to CVs which are quite similar to each other – as in this case, CVs from IT experts – is of greater practical usefulness because more fine-grained distinctions can be made. Third, projects in companies often require IT experts with specific competences and thus, this application context is highly important.

To obtain CVs from IT experts, we used a web crawler (cf. Section 3.1, c)). This choice was made to allow the acquisition of CVs from private homepages, which many IT freelancers maintain. The web crawler was fed with search terms commonly used in IT (e.g., "Java") and including "CV". Based on these search terms and starting from the *Google* search, the web crawler stored approximately 27,000 PDFs.

4.2 General Pre-processing

To ready the collection for text pre-processing, we had to resolve the challenges (C1) and (C2) described in Section 3.2. PDFs were stored in a MongoDB database and converted to TXT for further analyses.

In order to address (C1) the diverse languages present in the collection, we performed automatic language identification and partitioned the collection accordingly. To this end, we used a Java open source tool [35]. On a manually inspected test sample of 100 documents, the approach did not produce any errors. We proceeded with German documents as those represented the largest proportion of the collection.

With regard to (C2), we observed that a quite large percentage of documents were not actually CVs. Thus, we manually classified documents into CVs and non-CVs to obtain a training dataset and built a classification model based on majority voting of the common classification methods logistic regression, support vector classification and random forests. The classification into CVs and non-CVs reached an accuracy of 95% on a test dataset of 1,291 documents and was applied to the remainder of the collection.

Overall, general pre-processing resulted in a database of 2,410 (presumed) CVs in German language which we used for further analyses. An exploratory data analysis was performed to obtain an overview. For instance, we determined the number of total (3,504,014) and unique (242,416) terms in the database and analyzed which terms occurred most frequently (all of them common stop words for German texts).

4.3 Text Pre-processing

The pre-processing routines with special characteristics in regard to CVs were set up as follows: In order to determine a comprehensive stop word list, we started by integrating established lists [36, 37]. Then, we modified this list by incorporating CV-specific stop words based on own reflections as well as an analysis of the 1,500 most frequent words in the database. The CV-specific stop words included in particular time specifications, legal forms of organizations and forms of address. Numbers were filtered as well. In contrast, terms such as "C" or "R" were removed from the list, as they represent ability in the respective programming languages in the given context.

To create n-grams, we used the NPMI-based method from the Python topic modeling library gensim [38]. 2-grams were created for terms with a NPMI value of at least 0.5 and a joint occurrence frequency of at least 100. Analogously, 3-grams were created from 2-grams (e.g., "Microsoft Visual" and "Visual Studio" were combined to "Microsoft Visual Studio") and so on. In this way, many n-grams were created, the most frequently used ones being "SQL Server", "SAP R3" and "MS Office".

Part-of-speech tagging was realized by the aggregated results of two taggers. We used TreeTagger [39], a tagger based on a probabilistic Markov model pre-trained for the German language, and the Natural Language Toolkit [40] tagger which was trained with the TIGER corpus [41]. On a manually inspected test sample of 400 words, tagging in this way exhibited an accuracy of 96%. TreeTagger was also used for lemmatization.

For named entity recognition, the Stanford NER Tagger [42] was used. Its results were then refined by publicly available lists of first names [43, 44] and a phone book for surnames, achieving a true positive rate of 97% in regard to filtered person names.

As suggested in Section 3.3, we fixed many of the pre-processing routines but let others vary and experimented in order to achieve optimal results. To be more precise, we always – and in this order – removed formatting tags and special characters, tokenized and lowercased the CVs and removed stop words and words occurring only in few documents. We experimented with n-gram-creation (yes/no), lemmatization (yes/no), part-of-speech filtering (possibly filtering adjectives and/or verbs and/or adverbs) as well as the threshold for words to occur in too few documents (50/40/30/20/10). Overall, this resulted in 160 pre-processing configurations.

For each configuration, LDA models were generated using the gensim libary [38] and each number of topics $N \in \{25,50,75,100\}$, following [32]. Convergence was tested as suggested in [15]. Subsequently, the generated LDA models were evaluated with respect to semantic coherence (cf. Section 2.1). We determined the configuration leading to the highest value of semantic coherence and used it for optimizing the LDA application (cf. Section 4.4). Thereafter, to verify the results, we re-ran the evaluation of all configurations with the optimized LDA model. The configuration leading to the highest value of semantic coherence (NPMI: 0.161) did not consider n-gram-creation and lemmatization, filtered adjectives as well as verbs and adverbs, and filtered all words occurring in less than 40 documents.

The results in regard to n-gram-creation and lemmatization may surprise at first. However, they are in line with previous research in other application contexts suggesting that LDA derives required semantic relations itself and stronger pre-processing reduces topic model quality [30]. Similarly, filtering all parts-of-speech except nouns has also already been shown to provide strong results [29] and is expected to be promising for CVs, with competences usually being described by nouns.

4.4 Application of LDA

The hyperparameters α and β were optimized similar to the pre-processing configuration, again following [32]. To determine the number of topics N, we generated LDA models for each $N \in \{2,4,6,...,100\}$ based on the optimal pre-processing configuration. We then analyzed semantic coherence for each N. This led to choosing

N=42, a number of topics manageable for humans. Thus, the respective topic model was examined further with respect to interpretability and use of results.

4.5 Interpretation & Use of Results

The interpretation of the topic model was conducted separately by two human coders to account for human subjectivity. Consolidating the interpretation only required to settle minor wording differences. The word-topic-distributions of each topic were analyzed to obtain an overview of the topic model. We observed that topics could generally be categorized into three groups: Group A, the largest group, contained topics describing specific IT-related competences and consisted of 23 topics. The four topics in Group B described competences concerning business & management. The remaining 15 topics in Group C were related to university or school career and, due to our focus on competences, not considered for further analysis. Some topics are shown exemplarily in Table 1. Thereby, a topic is represented by its seven words with highest probability in decreasing order and translations for German words are provided in square brackets. Please note that many English words are used frequently in German CVs, explaining their occurrence in topics.

Table 1. Exemplary topics from each of the groups A, B, C

ID	*Gr.*	*Topic (most probable words)*
1	A	java eclipse entwicklung[development] xml spring j2ee oracle
2	A	linux server administration unix system perl security
3	A	design adobe konzeption[conception] 3d web programmierung[programming] photoshop
4	A	entwicklung[development] web javascript php mysql css html
5	A	windows server ms microsoft support software office
6	A	c r analysis time solution network networks
7	B	management projekt[project] einführung[launch] analyse[analysis] projektmanagement[project management] business durchführung[execution]
8	C	university research school international european science german

Overall, the analysis yielded that most of the topics in Groups A and B are fine-grained topics clearly representing specific competences. More precisely, they do not describe competences rather general for CVs from IT experts such as programming skills, but more distinguishing competences such as programming skills in Java (cf. Topic 1). Employing the relevance metric of LDAvis [34], it was possible to differentiate topics even further and carve out competences highly characteristic for a topic. Usually, this concerned closely associated special frameworks, software or technologies. For instance, the most relevant words of Topic 1 (describing programming skills in Java) then were: jaxb, j2se, jpa, jax, ejb, hibernate, jms. All of them are Java-specific and occurred almost exclusively in Topic 1.

The results support that the topics possess a clear interpretation with respect to

describing competences and are thus useful for HRM. Furthermore, based on these results, a topic-based search seemed promising. We labelled 21 of the 27 topics in Groups A and B with respect to the competence described (the remaining 6 topics were judged to not be as clearly interpretable and left out). Besides their use to rapidly assess the competences represented in a CV for HRM (cf. Section 3.5), the labels facilitated the topic-based search. Our prototypical implementation allows the specification of search queries as vectors containing the desired weight for each topic. Similarities between the search vector and the CV-topic-distributions of each CV are calculated based on Kullback-Leibler divergence [17]. The most similar CVs are shown together with their topic distribution. Figure 2 illustrates the GUI of the prototypical implementation with the search query from Section 3.5 and the first search result (CV #545). Clicking on a search result opens the respective CV.

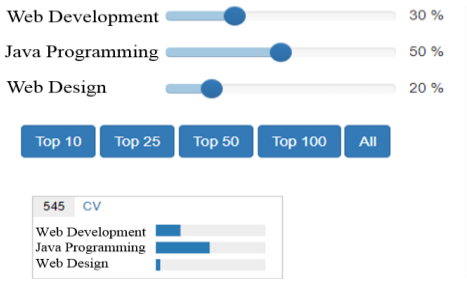

Figure 2. Topic-based search for CVs

CV #545 portrayed a senior Java developer with skills in a large number of Java-related technologies (Spring, Struts, JUnit, JEE, ...). The CV also claimed a lot of work expertise in web development such as the programming of web frontends using HTML and CSS. To a lesser extent, competences in web design and respective software (e.g., Photoshop, Gimp) were reported as well. Thus, the CV fit the job offer represented by the search query exceptionally well. The analysis of the other top 10 CVs which were determined to be the best match for the search query yielded similar results.

For comparison, we also performed a keyword search using the search term *java AND "web development" AND "web design"*. Here, we observed all 3 advantages of a topic-based search outlined in Section 3.5: With respect to 1), the keyword search only yielded 4 results because few CVs actually followed the exact wording dictated by the search term. In particular, many suitable CVs such as the ones found by our topic-based search were neglected by the keyword search. Regarding 2), the problem of merely focusing on keywords became obvious as the CV of a manager who once had conducted a Java project was included in the 4 results of the keyword search, but did in fact not fit the job offer. Concerning 3), the lack of weighting showed when the CV of a web developer specializing in PHP, HTML and JavaScript with basic Java abilities was assessed as relevant. Overall, none of the results of the keyword search fit the job offer well. Our topic-based search thus produced clearly superior results in this setting.

We further specified 8 more search queries and analyzed the respective CVs suggested by the topic-based search. In each case, the competences reported in the top

CVs coincided with the competences called for by the search query. To conclude, the topic-based search worked very well in this application and seems fit to provide helpful decision support for HRM.

5 Conclusion, Practical Implications and Directions for Future Work

In this paper, a topic modeling procedure consisting of five steps with the aim of discovering knowledge from CVs has been presented. CV-specific characteristics are considered in each step. An exemplary application to CVs from IT experts suggests that clearly interpretable topics describing fine-grained competences (e.g., Java programming, web design) can be discovered. This information can be used to rapidly assess the contents of a CV, categorize CVs and identify promising candidates for job offers, thus providing decision support in HRM.

The presented procedure allows for proactive recruiting. It can, for instance, be applied in HRM similar to how professional social networks are currently used in the recruitment process to rapidly source candidates before subsequent steps such as job interviews are conducted. However, it is not restricted to members of these networks as the analyzed CVs may stem from any origin. Additionally, the presented topic-based search possesses advantages compared to a keyword search on these platforms. Moreover, the CV-topic-distributions in conjunction with labels can be used to categorize and tag CVs for future use. In this way, companies can construct and steadily extend a database of interesting CVs. Another promising idea for companies is to also include CVs of own employees to promote internal recruiting. In any case, HRM and IT departments need to cooperate as skills from both areas are required to achieve a successful application of the procedure.

While the paper at hand offers a detailed description of a procedure for knowledge discovery from CVs, there are also limitations which provide directions for further research. First, an application to CVs from a different context should be conducted to validate feasibility. Second, a topic-based search technique has been presented and evaluated in an exemplary setting. However, it should be further assessed, for instance by a more detailed comparison to alternatives (e.g., with the help of a HRM expert) and an application to real job offers from a company. Finally, CV-specific visualization approaches should be developed, allowing for an easier overview and use of the results of the procedure. They should be included in a tool facilitating and partly automating the five steps of the procedure to further its practical use.

References

1. Breaugh, J.A.: Employee recruitment: Current knowledge and important areas for future research. Human Resource Management Review 18, 103–118 (2008)
2. Hendry, C.: Human resource management. Routledge (2012)
3. Allden, N., Harris, L.: Building a positive candidate experience: towards a networked model of e-recruitment. Journal of Business Strategy 34, 36–47 (2013)

4. Gao, L., Eldin, N.: Employers' expectations: A probabilistic text mining model. Procedia Engineering 85, 175–182 (2014)
5. Abel, F., Deldjoo, Y., Elahi, M., Kohlsdorf, D.: Recsys challenge 2017: Offline and online evaluation. In: Proceedings of the 11th ACM RecSys, pp. 372–373 (2017)
6. LinkedIn: About LinkedIn, https://about.linkedin.com/
7. Coffey, A., Atkinson, P.: Making sense of qualitative data: complementary research strategies. Sage Publications, Inc (1996)
8. Debortoli, S., Müller, O., Junglas, I., vom Brocke, J.: Text mining for information systems researchers: an annotated topic modeling tutorial. CAIS 39, 111–135 (2016)
9. Strohmeier, S., Piazza, F.: Domain driven data mining in human resource management: A review of current research. Expert Systems with Applications 40, 2410–2420 (2013)
10. Gupta, V., Lehal, G.S.: A survey of text mining techniques and applications. Journal of emerging technologies in web intelligence 1, 60–76 (2009)
11. Blei, D.M.: Probabilistic topic models. Communications of the ACM 55, 77–84 (2012)
12. Gorbacheva, E., Stein, A., Schmiedel, T., Müller, O.: The role of gender in business process management competence supply. BISE 58, 213–231 (2016)
13. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research 3, 993–1022 (2003)
14. Belford, M., Mac Namee, B., Greene, D.: Stability of topic modeling via matrix factorization. Expert Systems with Applications 91, 159–169 (2018)
15. Hoffman, M., Bach, F.R., Blei, D.M.: Online learning for latent dirichlet allocation. In: NIPS 23, pp. 856–864 (2010)
16. Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J.L., Blei, D.M.: Reading tea leaves: How humans interpret topic models. In: NIPS 22, pp. 288–296 (2009)
17. Koltcov, S., Koltsova, O., Nikolenko, S.: Latent dirichlet allocation: stability and applications to studies of user-generated content. In: Proceedings of the 6th ACM WebSci, pp. 161–165 (2014)
18. Aletras, N., Stevenson, M.: Evaluating topic coherence using distributional semantics. In: Proceedings of the 10th IWCS, pp. 13–22 (2013)
19. Lau, J.H., Newman, D., Baldwin, T.: Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In: Proceedings of the 14th EACL, pp. 530–539 (2014)
20. Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Proceedings of the 8th NAACL, pp. 100–108 (2010)
21. Röder, M., Both, A., Hinneburg, A.: Exploring the space of topic coherence measures. In: Proceedings of the 8th ACM WSDM, pp. 399–408 (2015)
22. Fang, D., Yang, H., Gao, B., Li, X.: Discovering research topics from library electronic references using latent Dirichlet allocation. Library Hi Tech (2018)
23. Guo, Y., Barnes, S.J., Jia, Q.: Mining meaning from online ratings and reviews: Tourist satisfaction analysis using latent dirichlet allocation. Tourism Management 59, 467–483 (2017)
24. Dumont, T., Fettke, P., Loos, P.: Towards multi-dimensional Clustering of Business Process Models using Latent Dirichlet Allocation. Tagungsband zur Multikonferenz Wirtschaftsinformatik, 69–80 (2016)
25. Shuyo, N.: Language detection library, https://www.slideshare.net/shuyo/language-detection-library-for-java
26. Jauhiainen, T., Lui, M., Zampieri, M., Baldwin, T., Lindén, K.: Automatic language Identification in texts: A survey. arXiv preprint arXiv:1804.08186 (2018)

27. Boyd-Graber, J., Mimno, D., Newman, D.: Care and feeding of topic models: Problems, diagnostics, and improvements. Handbook of mixed membership models and their applications, 225–255 (2014)
28. Weiss, S.M., Indurkhya, N., Zhang, T., Damerau, F.: Text mining: predictive methods for analyzing unstructured information. Springer Science & Business Media (2010)
29. Martin, F., Johnson, M.: More efficient topic modelling through a noun only approach. In: Proceedings of the ALTA 2015, pp. 111–115 (2015)
30. Schofield, A., Mimno, D.: Comparing apples to apple: The effects of stemmers on topic models. Transactions of the ACL 4, 287–300 (2016)
31. Spies, M.: Topic modelling with morphologically analyzed vocabularies. Scientific Publications of the State University of Novi Pazar Series A: Applied Mathematics, Informatics and mechanics 9, 1–18 (2017)
32. Wallach, H.M., Mimno, D.M., McCallum, A.: Rethinking LDA: Why priors matter. In: NIPS 22, pp. 1973–1981 (2009)
33. Chaney, A.J.-B., Blei, D.M.: Visualizing Topic Models. In: ICWSM (2012)
34. Sievert, C., Shirley, K.: LDAvis: A method for visualizing and interpreting topics. In: Proceedings of the workshop on interactive language learning, visualization, and interfaces, pp. 63–70 (2014)
35. Shuyo, N.: Language detection, https://github.com/shuyo/language-detection
36. Salton, G. and Buckley, C.: Stop word list 2, http://www.lextek.com/manuals/onix/stopwords2.html
37. Götze, M. and Geyer, S.: German stopwords, https://github.com/solariz/german_stopwords/blob/master/german_stopwords_full.txt
38. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50 (2010)
39. Schmid, H.: TreeTagger - a part-of-speech tagger for many languages, http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
40. NLTK Project: Natural Language Toolkit, https://www.nltk.org/
41. Institut für maschinelle Sprachverarbeitung: TIGER corpus, http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.html
42. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling. In: Proceedings of the 43rd ACL, pp. 363–370 (2005)
43. Michael, J.: Anredebestimmung anhand des Vornamens, https://www.heise.de/ct/ftp/07/17/182/
44. Kolb, P.: Liste mit Vornamen, http://www.ling.uni-potsdam.de/~kolb/Vornamen.txt