

# **Natural Language Processing on the Balance of the Swedish Software Industry and Higher Vocational Education**

Emil Bäckstrand and Rasmus Djupedal

Final Project

Main field of study: Computer Engineering BA (C)

Credits: 15

Semester/year: Spring 2023

Supervisor: Francisco Gomes de Oliveira Neto

Examiner: Felix Dobslaw

Course code/registration number: DT133G

Programme: Software Engineering

# Natural Language Processing on the Balance of the Swedish Software Industry and Higher Vocational Education

Emil Bäckstrand and Rasmus Djupedal

*Department of Communication, Quality Management and Information Systems*

*Mid Sweden University*

*Östersund, Sweden*

{emba2001, radj2000}@student.miun.se

**Abstract**—The Swedish software industry is fast-growing and in need of competent personnel, the education system is on the front line of producing qualified graduates to meet the job market demand. Reports and studies show there exists a gap between industry needs and what is taught in higher education, and that there is an undefined skills shortage leading to recruitment failures. This study explored the industry-education gap with a focus on higher vocational education (HVE) through the use of natural language processing (NLP) to ascertain the demands of the industry and what is taught in HVE. Using the authors' custom-made tool Vocational Education and Labour Market Analyser (VELMA), job ads and HVE curricula were collected from the Internet. Then analysed through the topic modelling process latent Dirichlet allocation (LDA) to classify lower-level keywords into cohesive categories for document frequency analysis. Findings show that a large number of HVE programmes collaborate with the industry via indirect financing and that job ads written in Swedish consist, in larger part, of inconsequential words compared to ads written in English. Moreover, An industry demand within cloud and embedded technologies, security engineers and software architects can be observed. Whereas, the findings from HVE curricula point to a focus on educating web developers and general object-oriented programming languages. While there are limitations in the topic modelling process, the authors conclude that there is a mismatch between what is taught in HVE programmes and industry demand. The skills identified to be lacking in HVE were associated with cloud-, embedded-, and security-related technologies together with architectural disciplines. The authors recommend future work with a focus on improving the topic modelling process and including curricula from general higher education.

**Index Terms**—Swedish Software Industry, Higher Vocational Education, Software Engineering, Latent Dirichlet Allocation, Document Frequency Analysis

## I. INTRODUCTION

With an ever-increasing digitisation of the economy and the industry, the Swedish software industry is in an expansive phase and needs to expand its personnel [1]. From the IT branch, voices are raised that companies are having problems finding qualified personnel [2], [3]. There is an increasing number of employers having this problem [4]. This situation puts a lot of stress on the industry as a whole, which in the short term leads companies to pass on job offers and prevents company growth. It also affects the staff, which has to pull

more weight to compensate for the lack of personnel. In the long run, it reduces the state's tax income and could impact the welfare system [4]. Therefore it is important to address the problem and identify the causes of these recruitment difficulties. Here, the education sector is on the front line of producing qualified graduates to meet the job market needs. However, universities have more stakeholders than just the industry to comply with when formulating their syllabi. Instead, there exists another post-secondary form of education, namely higher vocational education (HVE). HVE consists of both theoretical and practical studies and is provided in close cooperation with the industry. The main purpose of HVE is to supply the labour market with the right competencies. Therefore this thesis will be focused on HVE and analysing their curricula in accordance with the industry needs.

A limited number of studies are available that predominantly focus on the gap between industry needs and what is taught in higher education in Sweden. Investigative research and interviews established that universities cannot solely look at industry demand when designing syllabi and that quantitative research is warranted [5]. Furthermore, one study analysed circa 24,500 job ads together with a number of university syllabi with a custom-built tool called JMAR<sup>1</sup>. However, JMAR analyses the datasets based on predefined keywords mined from the Stack Overflow developer survey<sup>2</sup>, limiting the tool's capacity in capturing all relevant keywords [6]. Moreover, a survey with circa 4,600 company respondents found that graduates are lacking in both technical and software skills, and require supplementary training before starting a job [7], [8].

Previous work has primarily focused on general higher education, and to the best of the authors' knowledge, no study exists in the area of HVE and industry demand, consequently leaving a knowledge gap in the domain. This thesis will shine a light on that domain by collecting active job ads and HVE curricula in Sweden focused on software development and programming. The authors' created a tool called

<sup>1</sup><https://github.com/kristian-angelin/JMAR>

<sup>2</sup><https://insights.stackoverflow.com/survey/2021#overview>

Vocational Education and Labour Market Analyser (VELMA)<sup>3</sup> that automatically collects job ads and HVE curricula and clusters lower-level keywords to higher-level categories via topic modelling using a natural language process (NLP) called latent Dirichlet allocation (LDA). Finally, VELMA produce results from the modelling process and through document frequency analysis. The authors interpret these results to identify industry demands, and the technologies taught in HVE and subsequently identify the skills shortage in the Swedish software industry.

## II. PURPOSE AND CONTRIBUTIONS

The Swedish software industry is fast growing, and the demand for software developers is constantly increasing [9]. Furthermore, there is an undefined skills shortage leading to recruitment failures [1], [2], [4], [10].

This study will expand upon previous work and extend it by including larger data sets and looking into HVE curricula to investigate and analyse the skills mismatch between education and the industry. VELMA will generate quantitative data by automatically fetching HVE curricula and active job ads and then compare these by clustering concrete keywords into categories with the NLP process LDA. For example, keywords such as CSS, HTML, and JavaScript are grouped into topics which can be named by high-level terms like “Frontend” or “Web”. Finally, the topics’ keywords will be used for document frequency analysis to ascertain what technologies are in demand by the industry and which are taught in HVE programmes. However, one risk with NLP is the noise generated by commonplace words irrelevant to the domain, such as *anställd* (employee) or *studenten* (student). Further investigation is needed to assess how much of an impact the noise has on the modelling process and whether the data needs to be filtered to remove the noise.

The unique contribution of this paper is (i) to analyse the skills shortage with a focus on HVE, (ii) to provide a tool that will automatically collect and cluster keywords into topics in active job ads and HVE curricula, (iii) to lay the groundwork for future NLP studies on Swedish job ads and curricula and finally, (iv) compare job ads and HVE curricula by producing quantitative results via document frequency analysis.

The research questions that will be leading the work in this study are:

- **RQ1:** What is the identifiable noise when conducting topic modelling on curricula and job ads?
- **RQ2:** What technologies are in high demand in the Swedish software industry today?
- **RQ3:** What technologies are delivered by HVE today?
- **RQ4:** Are there technologies and skills in demand by the industry that are not being delivered in higher vocational education?

<sup>3</sup><https://github.com/embaradj/VELMA>

## III. BACKGROUND

### A. Higher Vocational Education

The idea behind the Swedish HVE system is to continuously supply the industry with competent labour while being able to adapt quickly in accordance with shifting industry demand. The length of HVE programmes varies, but the majority of those within Software Engineering (SE) last for about two years, which is also the minimum duration required to issue an *kvalificerad yrkeshögskoleexamen* (Advanced Higher Education Diploma)<sup>4</sup>. *Myndigheten för yrkeshögskolan* (The Swedish National Agency for Higher Vocational Education) or MYH<sup>5</sup>, is the authority responsible for issuing permits to start and run HVE programmes. MYH receives funding from the parliament yearly and then decides which HVE programmes to permit, and the number of students each programme can have. There are several criteria that MYH needs to take into account before approving an HVE. One of these is that there must exist an industry demand within the field of HVE education, or, be in great personal or public interest<sup>6</sup>. It is not enough that just one company has a demand of certain labour. Both public and private organisers can provide HVE programmes, but there must always be a management group consisting of members from the industry behind the application<sup>7</sup>. Most of the HVE programmes receive state funding, and all of these are free of charge for the students. Generally, they are also financed by the industry. MYH promotes education with a higher grade of industry financing and takes this into account when considering several competing applications to start an HVE programme<sup>8</sup>.

An important part of HVE is *lärande i arbete* (learning in a work environment) or LIA, which is a form of an internship at a company. This in combination with the theoretical knowledge forms valuable experiences sought by the industry. Companies which are engaged in the HVE programmes have both the ability to influence the theoretical parts of education, but also the ability to form the students during their internship. The internship must have a duration of at least half a year to issue an advanced higher education diploma<sup>9</sup>. Statistics from MYH show that about 50% of HVE graduates start working at a company where they have previously conducted their internship<sup>10</sup>. HVE programmes do not have a central admission system, as opposed to Swedish colleges and universities. Instead, it is up to the HVE organiser to decide

<sup>4</sup><https://www.myh.se/yrkeshogskolan/for-utbildningsanordare/larande-i-arbete-lia>

<sup>5</sup><https://www.myh.se/in-english>

<sup>6</sup>[https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/forordning-2009130-om-yrkeshogskolan\\_sfs-2009-130#K1](https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/forordning-2009130-om-yrkeshogskolan_sfs-2009-130#K1)

<sup>7</sup>[https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/forordning-2009130-om-yrkeshogskolan\\_sfs-2009-130#K4](https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/forordning-2009130-om-yrkeshogskolan_sfs-2009-130#K4)

<sup>8</sup><https://www.myh.se/yrkeshogskolan/ansok-om-att-bedriva-utbildning/kort-om-forutsattningar-och-krav>

<sup>9</sup><https://www.myh.se/yrkeshogskolan/for-utbildningsanordare/larande-i-arbete-lia>

<sup>10</sup><https://www.yrkeshogskolan.se/antagning-och-studier/larande-i-arbete-LIA/>

the requirements for their candidates. This allows for some flexibility and can provide early selection, which can help to reduce the number of dropouts.

All HVE programmes in Sweden are classified via the Swedish Standard Classification of Education (SUN)<sup>11</sup>. The SUN-code specifying the orientation of education is built on three digits and one letter in the following format; *481a*. Where the two starting digits specify the main orientation, the third the subject orientation and the letter the specification. The code *481a* implies education with a focus on system development and programming.

### B. Job market

*Arbetsförmedlingen* (The Swedish Public Employment Service)<sup>12</sup> is responsible for the labour market in Sweden. This includes not only matching employees with people looking for work, but also providing informational research, the future prognosis of the labour market, and analysis of its current state. JobTech Development<sup>13</sup> is an open ecosystem with backing from Arbetsförmedlingen with 200 companies and organisations working on the project. It includes several components, from open data sets to ready-made applications and open APIs. One component of particular interest is JobStream API<sup>14</sup>, which allows for collecting active job ads. All job ads fetchable through JobStream are classified via the Swedish Standard Classification of Occupation (SSYK)<sup>15</sup>. The occupational ID is made up of four digits in the following format; *2512*, where the first digit specifies the professional field, the second the main group, the third the occupational group and the fourth the subgroup. The occupational id *2512* implies an occupation in the IT field requiring a higher education with a focus on software and system development.

### C. Topic modelling

Topic modelling is an unsupervised method for generating a statistical model of a dataset in the realm of NLP. This procedure is useful when there is a need to analyze many text documents to discover what the documents constitute. There are several methods for conducting topic modelling, where the models are more or less suited for different types of datasets. One such model is LDA, where an overview of LDA can be seen in Figure 1. LDA will look through a corpus, where the corpus is a pre-processed dataset and assigns words to  $K$  number of topics, over  $N$  number of iterations. Where words in each topic will be distributed based on a probability factor, however, because of the unsupervised nature of LDA, there is an issue with noise in the data. As such, noise in the form of irrelevant words to the topics of interest will have to be filtered out. This can be done either in the form of manually adding identified noise to a stopwords file or by way of stemming and

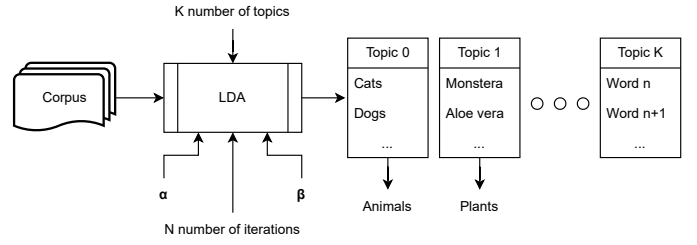


Fig. 1: LDA overview.

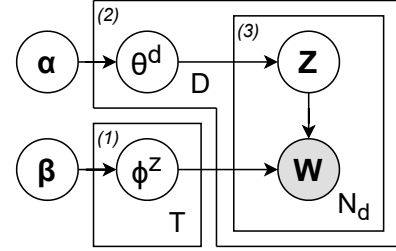


Fig. 2: Plate notation of the LDA model.

lemmatisation, which analyses the meaning behind words. The purpose of stemming and lemmatisation is to reduce derivative words to their root form i.e., “studying” to “study”. Stemming is the simpler process, where the ends of words are removed e.g., “tool’s” to “tool”. Lemmatisation on the other hand tries to uncover the meaning of words by grouping derivative words into a single word, also called the *lemma*. However, the effects of stemmers are ambiguous [11], and will not be explored further in this study.

Moreover, LDA is a sampling-based model, utilising the *Gibbs sampling* algorithm [12]. Figure 2 illustrates the LDA model with the sampling steps. The variable  $\alpha$  is responsible for the Dirichlet distribution regarding the topic dispersal on each document. The  $\alpha$  value is a prior, i.e., it tells the model the likelihood of topic distribution per document, where a higher value indicates documents are likely to include a higher number of topics. The other prior  $\beta$  defines the likelihood of word distribution for each topic, where a higher value indicates a topic is likely to comprise a greater mixture of words. Whereas,  $\theta$  is the topic distribution,  $\phi$  is the word distribution,  $\mathbf{Z}$  is the topic identity of all words and  $\mathbf{W}$  is the identity of all words. The variables can either be observed or unobserved, i.e. *latent*, indicated by the shading, where  $\mathbf{W}$  is an observed variable of a given word. The rest of the unshaded variables are latent, and not known until the modelling process has begun. Each plate (represented by the boxes) emphasises the repeated sampling of topics and words. Plate 1 illustrates the word distribution sampling for every topic  $z$  until  $T$  number of topics have been generated. Furthermore, plate 2 illustrates the topic distribution sampling for each document  $d$  until a total of  $D$  number of documents. Finally, plate 3 illustrates the topic and word sampling until  $N_d$  words have been generated for the document  $d$ .

<sup>11</sup><https://sun2020.scb.se/home>

<sup>12</sup><https://arbetsformedlingen.se/other-languages/english-engelska>

<sup>13</sup><https://jobtechdev.se/en>

<sup>14</sup><https://jobstream.api.jobtechdev.se/>

<sup>15</sup><https://www.scb.se/dokumentation/klassifikation-och-standarder/standard-for-svensk-yrkesklassificering-ssyk/>

#### IV. RELATED WORK

Bodell [5] investigated the syllabi of higher education in contrast to the needs of the industry, together with conducting interviews with employees from a number of companies within SE. Bodell concluded that while the different educations explored respond well to industry demands, considerations should be made regarding practical work and internship when updating syllabi. While at the same time, there are more actors to consider than solely the needs of the industry when designing higher education syllabi. Finally, Bodell recommends future work in the realm of quantitative studies. This study aims to address this recommendation and explore the balance of the Swedish software industry and HVE through quantitative methods.

Dobslaw et al. [6] on the other hand conducted a quantitative analysis comparing keywords of a large number of job ads with the syllabi from a number of higher educations. The job ads were collected through the authors' custom-created tool JMAR<sup>16</sup>. A total of 24,498 job ads were collected, which were compared to syllabi from 17 educational programs. The authors concluded that while some discrepancies could be observed, the method of analysing the supply and demand of technical skills was insufficient to warrant any changes to the explored syllabi. The authors state that neither recruiters nor the people responsible for formulating syllabi at higher education likely follow the CC2020 classifications, which were used when selecting the syllabi explored in the study. Moreover, the authors recommend future work in the same realm but with a focus on different types of SE skills, investigating broader programming concepts and utilising NLP. This study aims to expand on the work of Dobslaw et al. in several ways, (i) by exploring HVE instead of general higher education because of HVE purpose of supplying the labour market with the right competencies, (ii) collect HVE curricula and job ads programmatically with a focus on system development and programming, (iii) utilise NLP to generate topics for document frequency analysis to explore the Swedish industry-education gap.

Furthermore, Andersson et al. [7] developed a study in collaboration with SWEDSOFT<sup>17</sup> and *Statistiska centralbyrån* (Statistics Sweden), or SCB<sup>18</sup> where 4,598 companies responded to a survey regarding questions surrounding software development and usage of software in general. The survey included questions concerning the competence of recent graduates, what kind of competencies was most sought after, and which they found most lacking. The survey was produced and validated by SWEDSOFT and SCB and a report by Andersson was also published by *Institutet för Näringslivsforskning* (The Research Institute of Industrial Economics) [8]. Results from the survey concluded that graduates require supplementary training before they can take on a job. Moreover, A majority of the companies found graduates are lacking in technical and

software skills, together with organisation working methods. The study explored the industry-education gap from the view of the industry, where this study aims to look into the gap from both sides. Additionally, the study looked at the technical and software skills at an abstract level, i.e., no mention of specific technologies, where this study aims to focus on technologies and skills.

Finally, Gurcan and Kose [13] utilised LDA on 2,533 job ads within SE from Stack Overflow Careers<sup>19</sup> (now defunct) to identify industry demand and trends. The study used Mallet's [14] CLI interface to generate topics on roles and their responsibilities and the respective distribution in the dataset. Furthermore, the most popular combination of programming languages was explored, together with the education requirements and trending topics. The authors raise the issue of multilingual job ads and the issue of finding optimal parameters for the LDA process. Gurcan and Kose did not focus on the Swedish software industry, however, their usage of LDA, and more specifically Mallet, has a strong resemblance with this study. This study aims to complement the work of Gurcan and Kose by focusing on the Swedish software industry, but also including HVE curricula and exploring the noise generated by the respective datasets.

#### V. RESEARCH METHODOLOGY

This study's methodology can be divided into four sections; (a) artefact creation, (b) data collection, (c) data pre-processing, and (d) data analysing. An overview of the methodology can be observed in Figure 3.

##### A. Artefact creation

VELMA was created to automate the process of gathering both HVE curricula and job ads and then analysing the content. VELMA utilizes three different APIs in order to get the necessary information. Its GUI interface consists mainly of two scrollable lists, which get populated by HVE programmes and job ads when the corresponding search is run. The two buttons "Search Curriculum" and "Search job Ads" initiate the fetching of data from the APIs. Items that are populated in the HVE- or the job ads list are clickable, and clicking one of them opens a new window showing more details about the specific HVE or job ad. In the Settings menu, the filter parameters for the job ads search and parameters related to the analysis process can be adjusted. After searches have been executed for both HVE and job ads the analysing processes can be initiated by clicking the "Analyse" button. A process consisting of several steps starts, beginning with the topic modelling to generate categories of words, followed by topic selection and naming. Finally, keyword analysis of both datasets where a comparison is done based on the found words in the modelling process. Finally, after the analysis, the results are presented in a new window and saved to disk. An overview of the VELMA workflow can be observed in Figure 4, together with an overview of the GUI in Figure 5.

<sup>16</sup><https://github.com/kristian-angelin/JMAR>

<sup>17</sup><https://www.swedsoft.se/en/>

<sup>18</sup><https://www.scb.se/en/>

<sup>19</sup><http://careers.stackoverflow.com>

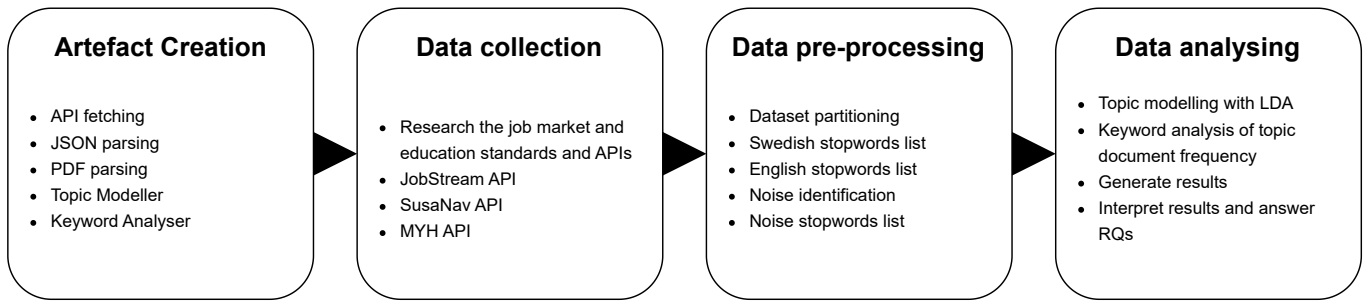


Fig. 3: Methodology overview.

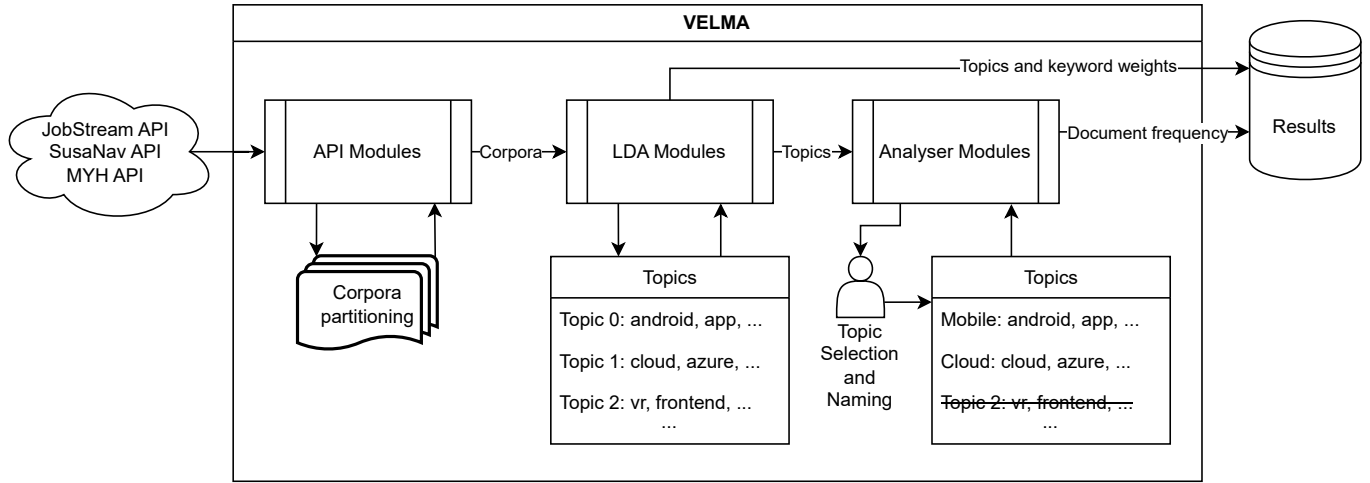


Fig. 4: Overview of the workflow of VELMA.

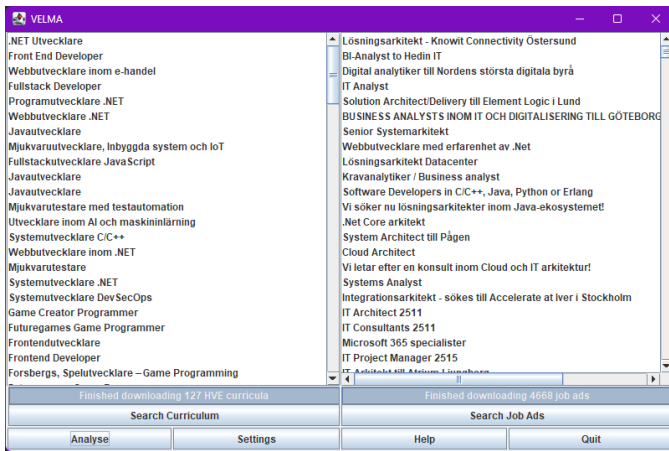


Fig. 5: Main window of VELMA.

### B. Data collecting

The process of data collecting consists of two parts, the HVE curricula collecting, and the job ads collecting. The data was collected on 2023-03-27 and the number of HVE curricula and job ads collected were 127 and 4,749, respectively.

HVE curricula are collected in two steps. First, SusaNav

API<sup>20</sup> is used to get a list of all the HVE codes within development and programming. This is done by including a parameter to the API request based on the SUN-code 481a, which specifies education with a focus on system development and programming. Each HVE code in the API response generates a second API request to MYH API<sup>21</sup>, which returns a list of all applications made to start a programme for that specific HVE code. The application is a document based on a common template containing all the courses and learning outcomes of the programme. VELMA picks the latest approved curricula for each programme and the URL of the curricula in PDF format is then parsed and downloaded. There is no available information about the existence of MYH API and therefore no documentation is available. To identify its capabilities and understand how to use it, the authors studied the code of the MYH webpage and the structure of its API requests. Finally, to utilise the API, VELMA was implemented to mimic the API requests of the MYH website's frontend.

Active job ads are collected through the JobStream API<sup>22</sup> based on the occupational ID set by SSYK. The occupational IDs of interest are in the range of 2,511–2,519, which specifies

<sup>20</sup><https://susanavet2.skolverket.se/#/api/>

<sup>21</sup><https://w3d3-integration-service.myh.se/1.0/search>

<sup>22</sup><https://jobstream.api.jobtechdev.se/>

occupations within different areas of system development. For instance, 2512 and 2516, denoting software- and system developers and IT-security specialists, respectively. This will ensure that both the HVE curricula and job ads collected will be appropriate, i.e., graduates from these HVE programmes would explore these job ads.

### C. Data pre-processing

The data is divided into a number of corpora such that the modelling process and the keyword analysis can be run on isolated datasets. Since there are a large number of job ads written in English on the Swedish labour market, ads are separated by the language of the documents into Swedish and English corpora during the API fetching process. Whereas HVE curricula are divided into three corpora consisting of the full curricula, aims of the programme and the included courses during the PDF parsing phase. By manually reviewing the curricula, the authors found that the subsection covering aims of the programme and included courses comprises the most information regarding the technologies taught by HVE.

The language filter for the job ads works by dividing the number of words in each ad containing the Swedish letters *å*, *ä* and *ö* with the ad's total number of words. If the job ads text consists of at least 5% Swedish words the ad is marked as Swedish. Verification of the filter's success rate was done by manually looking at 10% of a random selection of job ads written in Swedish and English where a failure rate of less than 1% could be observed. With the job ads dataset split, the resulting number of ads written in the respective language is 3,062 Swedish and 1,687 English.

The HVE dataset is split by collecting the curricula text as lines in an array. Then, methods for getting the index in the array of a specified text and returning the text at an index are used. The curricula were manually explored to identify the sections of interest, and between which lines to collect the text. Verification of the process was done by manually looking at 10% of the full curricula text and comparing it to the corresponding aim and courses corpus. An overview of the modules responsible for this process and the different corpora can be seen in Figure 6

For the unsupervised topic modelling to succeed in producing effective categories with cohesive keywords the corpora need to be pre-processed in several ways. First, a set of stopwords for commonly used words has to be included, as the job ads dataset consists of ads written in Swedish and English two lists of stopwords were included. The lists were collected from two websites<sup>23,24</sup> where the lists of Swedish and English respectively were merged from both websites, with duplicates removed. Examples of the stopwords from the respective list are *viktig* (important), *adjö* (goodbye), *should* and *or*.

Next, the trivial words undermining topic stability, called noise, were identified and subsequently added to a stopwords list. This was executed in several steps for identifying the noise and building a stopwords list to remove the noise.

<sup>23</sup><https://countwordsfree.com/stopwords>

<sup>24</sup><http://alir3z4.github.io/stop-words/>

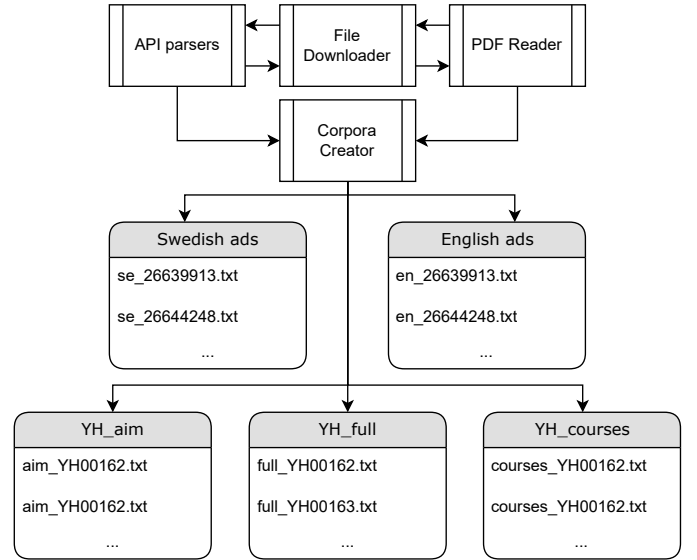


Fig. 6: API modules and dataset partitioning overview, YH is the acronym for *yrkeshögskola* (HVE) and corresponds to the respective HVE dataset.

TABLE I: LDA settings used when generating the stopwords file.

$\alpha$	0.01
$\beta$	0.01
No. topics	5
No. words	7
Threads	16
Iterations	2,000

**Identifying and Removing the noise:** Text files were created programmatically, by counting the number of times each word is found in the corpora, and in how many documents the word was found. A file was generated for each dataset corpus and is available on the study's data repository<sup>25</sup>. Copies of each file were sorted in descending order based on the number of hits.

The initial stopwords list was created by running the algorithm on the full dataset, and manually adding the trivial words identified to the stopwords file. Any uncertain words were left off the stopwords list, to be further explored if they impacted the modelling process. A seed value of **42** was used to reduce randomness between runs. The settings used to obtain these results are shown in Table I.

Each dataset's noise was explored in five iterations, where after each iteration identified noise was added to the stopwords file. After five iterations on each dataset, the procedure started over from the last dataset run in reverse order until no additional noise could be observed. The number of iterations needed, together with the order in which the dataset's noise was explored during this procedure is from left to right as observed in Table II.

<sup>25</sup><https://github.com/embaradj/thesisdata>

TABLE II: Number of iterations needed to remove noise, the columns denoted with YH corresponds to the different HVE datasets, Eng & Swe for the job ads and full for the whole corpora (YH\_full + Eng + Swe).

Dataset	YH_courses	YH_aim	YH_full	Eng	Swe	Full
Iterations	5	5	20	16	58	1

#### D. Data analysis

The data is analysed in two parts; (i) topic modelling with LDA and (ii) keyword analysis based on the found topics.

**Topic modelling:** The authors wanted to dynamically extract clusters of keywords to remove the dependency of a static keywords list of technologies. The reason is that a static list needs continuous updates to include the latest technologies. Moreover, creating an inclusion list containing relevant technologies introduces bias as it is impossible to ascertain that all information has been included. As such, the authors investigated topic modelling which includes a multitude of different models. Lee et al. [15] compared four models; latent semantic analysis (LSA), probabilistic latent semantic analysis (PLSA), latent Dirichlet allocation (LDA) and correlated topic model (CTM). The findings show that LSA and PLSA are best used when the documents comprise a single topic. Moreover, LDA outperforms the aforementioned models on lengthy documents and on documents with multiple topics. Where CTM, which is a hierarchical extension of LDA, was found to also allow for identifying the relationship between topics. However, topic-relationship is not necessary when conducting document frequency analysis. With that said, there are novel models that outperform LDA, such as hierarchical latent tree models (HLTMs), more specifically hierarchical latent tree analysis (HLTA) [16]. Despite this, the increased complexity of HLTMs, their verbose output and collecting the information required for conducting document frequency analysis i.e., grouped technology terms can be achieved with LDA. The authors determined that LDA is a good fit for this study's scope. Additionally, the favourable results of the topic modelling process in Gurcan and Kose [13] study which utilised LDA on job ads to find needs and trends in the software industry further strengthens the authors' model selection. Furthermore, while LDA is one of the simpler models [17] and is outperformed by some novel models, it is still one of the most popular and widely used models [18]. LDA and models extending on LDA have been used for a multitude of applications over the years, ranging from identifying coupling in object-oriented code [19], and classifying malicious applications in the Android operating system [20] to categorising consumer complaints of financial services [21]. Finally, LDA is extendable to more novel models such as hierarchical latent Dirichlet allocation (hLDA) [22], which opens up future work and improvements on this study.

The LDA model used in this study utilises the third-party API, Mallet [14] which has many well-optimized algorithms

TABLE III: LDA settings used when answering RQs for the respective dataset. HVE corresponds to the YH\_aim + YH\_courses corpora.

Dataset	$\alpha$	$\beta$	No. topics	No. words	Threads	Iterations
HVE	9.0	0.25	6	4	16	2000
Swe	8.0	0.25	5	4	16	2000
Eng	9.0	0.25	5	4	16	2000

for NLP, with the possibility to extend on and build new models if the need arises. This study utilises the *Parallel-TopicModel* class to incorporate a multithreaded sampling-based implementation of LDA, together with the *pipe* package to build the pipeline to pre-process the data. The model recursively looks through the corpora folders and adds the documents to the pipeline based on the program settings via a filter class. A number of parameters can be adjusted for the modelling process, where the values for answering **RQ2**, **RQ3** and subsequently **RQ4** can be observed for the respective dataset in Table III.

The optimal settings for each dataset were calculated using Mallets *MarginalProbEstimator* which is based on Wallach's *Left-to-right* algorithm [23]. The evaluation of hyperparameters and the number of topics are done programmatically by splitting the dataset into 80% training and 20% testing. Furthermore, the model runs on the combination of several  $\alpha$  and  $\beta$  values, together with an iterative process from 0–10 topics. This in turn results in a total of 2,560 combinations of varying parameters which are saved to file for the respective dataset, available in the study's data repository. The value evaluated is the probability of held-out documents also called log-likelihood, where a higher value indicates better fitness of the model on the dataset. The authors decided the optimal number of words by manually running the modelling and keyword analysis process on the optimal settings with word ranges from 3–7 and investigating the results. Then the topics were evaluated and the optimal number of words was chosen based on where there was enough detail to identify the topic, but not too verbose to constitute terms outside of the topic's scope.

When the topic modelling process finishes, a file is created containing the topics, the respective alpha value for the topic and the words with their respective weight. The topic alpha value is the Dirichlet parameter for the topic, a larger value indicates that a topic contains words that are frequent in many documents. Whereas a smaller value indicates that the topic consists of words found in a smaller set of documents. The word weight represents the probability of the word to be associated with the topic, a larger value means that the word is more likely to belong to the topic. Throughout the evaluation and topic modelling process the seed value of **42** was used to reduce randomness.

**Keyword analysis:** After the topic modeller has finished, it is possible to name the topics, as they are not being named automatically during the topic modeller process. Furthermore,

unwanted topics can be deselected in order not to be included in the keyword analysis. The analysis process is initiated when the topic naming and selection are finished. This is done by using document frequency. For each topic, the number of documents in which all of the topic's keywords are present is counted. This ensures that the respective document comprises the whole topic and not only specific keywords. The analyser only counts documents belonging to the datasets that are selected in the program settings. In order to produce results for **RQ2**, **RQ3** and subsequently **RQ4**, VELMA was run three times to produce individual topic modelling- and analysis results on the three datasets; Jobs (Swedish & English) and HVE (Aim & Courses). HVE aim and courses are treated as one sub-dataset by VELMA as the texts are extracted from the same curricula. After the analyser finishes the results are presented in a new window, as well as written to a file in order to simplify exporting the results. The results consist of a table with the topics and the document frequency.

## VI. RESULTS

### A. **RQ1:** What is the identifiable noise when conducting topic modelling on curricula and job ads?

A total of 127 HVE curricula and 4,749 job ads were collected, analysed and files generated programmatically to produce results of the identifiable noise. Figure 7 shows the top 10 noise in terms of word counts in each dataset. Also included in the figures is the document distribution of each word in the respective dataset. Most of the noise identified in the HVE dataset corresponds to typical words regarding education, such as *kunskaper* (knowledge) and *studerande* (undergraduate). Where most words can be found in the majority of the documents, one outlier is *medfinansiering* (co-financing). This indicates a large number of programmes receive financing from other actors outside of state financing. This in turn can correlate to a larger collaboration between the industry and education via indirect financing<sup>26</sup> from companies cooperating with institutions. By reviewing curricula containing the term *medfinansiering* the authors found this collaboration takes on many forms, such as lectures from consultants in the industry, study visits, learning in a work environment (LIA), and student industry meetings.

The majority of the noise in the job ads consists of words correlating with recruiting buzzwords, such as *team*, *erfarenhet* (experience) and *work*. Here, a difference can be observed where the words in the ads written in English are more evenly distributed among the dataset. Moreover, the majority of the words in the top 10 noise in the English dataset are analogous to recruitment. Whereas, in the top 10 noise in the Swedish dataset more words are noisier as in less relevant to the recruitment process.

<sup>26</sup><https://minasidor.myh.se/hjalp-och-information/guider/fragor-till-ansokan/#stycke11>

### **RQ1:** What is the identifiable noise when conducting topic modelling on curricula and job ads?

Noise in the curricula are found in the majority of the documents, with trivial words such as, *studerande* (undergraduate), *kunskaper* (knowledge) and *ansökan* (application). One trivial word, *medfinansiering* (co-financing), indicates increased collaboration with the industry where reviewing the curricula shows this takes on many forms, including lectures from consultants and student visits, among others.

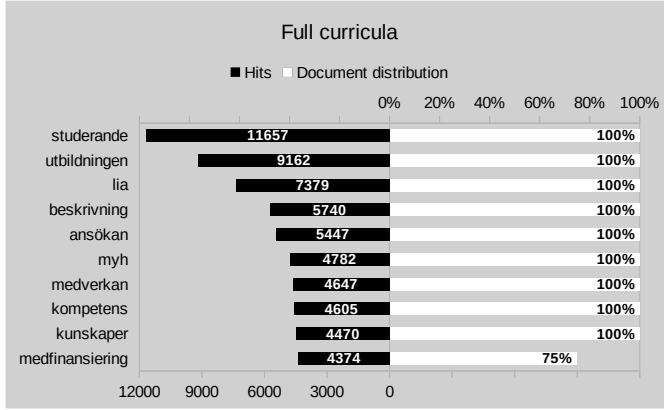
The job ads noise consists in large part of recruitment buzzwords such as, *erfarenhet* (experience), *team* and *skills*. Moreover, ads written in Swedish are noisier than those written in English.

### B. **RQ2:** What technologies are in high demand in the Swedish software industry today?

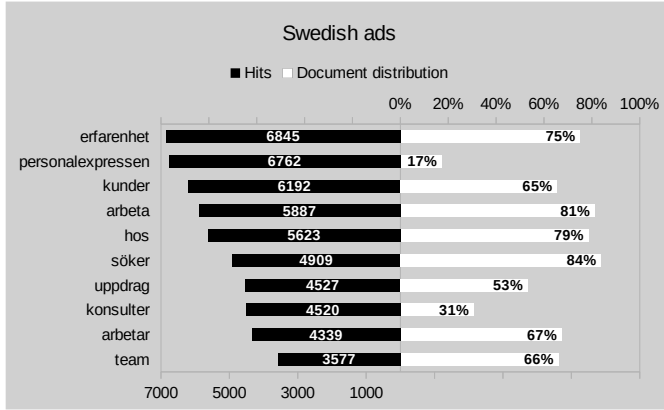
In order to answer the research question the authors look at the results VELMA produces when running the datasets “Swedish job ads” and “English job ads” respectively. These are presented in Table IV and Table V. The Swedish dataset contains 3,062 documents while the English contains 1,687 documents. There are some similarities among the topics generated, and by inspecting the keywords and the weight of each keyword in Figs. 8b and 8c, the authors manually identify and group the similar ones together over the datasets. Topic 0 of both datasets are categorised as *embedded*, topic 2 of both the datasets as *cloud*, and finally topic 4 of the Swedish dataset together with topic 3 of the English both being categorised as *web*. Additionally, topic 3 of the Swedish dataset is categorised as *architect* because of the abundance of terms related to architecture. Furthermore, topic 1 of the English dataset correlates with the category *security*, but the authors do not see such a topic in the Swedish dataset. The reasoning for this categorisation is as follows;

- **Embedded:** C++ and Python are both languages which are often found in embedded programming. While C++ provides low-level features, Python is heavily used in IoT applications and microcontrollers. Furthermore, as the results show, the words “embedded” and the Swedish equivalent “inbyggda” are both clustered together in these topics.
- **Cloud:** Both the grouped topics in the datasets share the keyword *cloud* while containing other words also often associated with cloud such as *bi*, which is a short for “business intelligence”, and *automation*.
- **Web:** Both the grouped topics contain web-related keywords such as *react*, *web* and *backend*. However, it also has some similarities with the *cloud* category, such as the keywords “cloud” and “agila (agile)”.

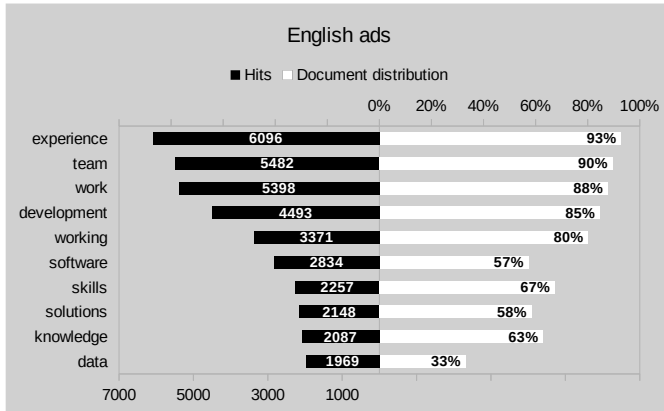
Topic 1 of the English dataset was categorised as **security** because its keyword *security* has a very high weight, as seen in Figure 8c. Compared with the English dataset, the Swedish has a more evenly distributed word weight among its topics



(a) Full curricula of HVE totalling to 127 documents.



(b) Swedish job ads totalling to 3,062 documents.



(c) English job ads totalling to 1,687 documents.

Fig. 7: Top 10 noise identified in the different datasets. The black bars on the x-axis comprise the number of occurrences in the respective dataset and the white bars the document distribution of the words.

TABLE IV: Topic document frequency in the Swedish job ads, totalling to 3,062 documents.

Topic	Category	Topic $\alpha$	No. Documents
0	Embedded	0.09	18
1	Null	0.05	0
2	Cloud	0.08	1
3	Architect	0.06	151
4	Web	0.12	26

TABLE V: Topic document frequency in the English job ads, totalling to 1,687 documents.

Topic	Category	Topic $\alpha$	No. Documents
0	Embedded	0.13	25
1	Security	0.18	25
2	Cloud	0.11	26
3	Web	0.19	17
4	Null	0.06	0

8b. The topics *topic 1* from the Swedish dataset and *topic 4* from the English dataset are ignored because their keywords originate from varied documents, as seen in Tables IV and V, i.e. there is no document containing all of the respective topic's keywords.

#### RQ2: What technologies are in high demand in the Swedish software industry today?

There is an industry demand within the fields of cloud- and embedded development, security, and for software architects, with keywords such as *bi*, *C++* and *python*. However, the demand for security-related knowledge was identified only in the English job ads.

#### C. RQ3: What technologies are delivered by HVE today?

127 HVE curricula were analysed to produce 6 topics with the 4 top-weighted words as seen in Figure 8a. Furthermore, through document frequency analysis the topics document occurrence was mapped and can be observed in Table VI. The majority of the topics are related to web development, more specifically topics 2, 3 and 5. The topics contain terms often found in relation to web development, such as; *javascript*, *frontend*, and *api*. The high number of document occurrences of the topics; 24, 51 and 9, respectively, in contrast to the number of curricula indicates that technologies related to web development are popular among HVE. Moreover, *topic 1* contain terms related to object-oriented programming (OOP), and *topic 4* contains terms related to game development. However, the document occurrence of *topic 4* is too small to draw any conclusion. The final topic, 0 has no occurrences in the dataset, indicating a badly generated topic in the modelling process.

TABLE VI: Topic document frequency in the HVE dataset, totalling to 127 documents.

Topic	Category	Topic $\alpha$	No. Documents
0	Null	0.05	0
1	OOP	0.16	7
2	Web 1	0.08	24
3	Web 2	0.11	51
4	Game	0.05	1
5	Web 3	0.10	9

**RQ3: What technologies are delivered by HVE today?**

HVE are mostly focused on web development and object-oriented programming with keywords such as, *javascript, frontend, java* and *C#*.

D. **RQ4: Are there technologies and skills in demand by the industry that are not being delivered in higher vocational education?**

By investigating the topic document frequency in **RQ2** & **RQ3** an education-industry mismatch can be observed. Most notable is the lack of cloud- and embedded-related technologies in the topics generated from the HVE dataset, both of which occur in respective job ads topics. Reviewing the job ads revealed some merits to be in demand in cloud- and embedded-related job posts. Below are a few examples of those merits:

**Cloud**

- *Deep knowledge within at least one of Amazon Web Services, Microsoft Azure or Google Cloud Platform*
- *Programming in common languages like Java, C#, JavaScript*
- *Serverless development e.g. using Lambda or Azure Functions*
- *Developing microservice solutions*
- *Developing Infrastructure as Code with technologies such as Ansible and Terraform*
- *Familiar with Linux, CI/CD pipelines, Docker and Kubernetes or other container technologies*
- *Build tools like Jenkins, Git*
- *Agile methodology*

**Embedded**

- *C/C++*
- *Embedded Linux*
- *Real-time OS, multithreading*
- *Git and Gerrit*

Furthermore, the word *architect* does not occur in any topic generated from the HVE dataset, but in both job- datasets, which indicates an unmet industry need for software architects.

Moreover, a security-related topic is found in the English dataset, whereas no analogous topic can be observed in the HVE topics.

**RQ4: Are there technologies and skills in demand by the industry that are not being delivered in higher vocational education?**

Cloud-, embedded-, and security-related technologies together with software architects are in demand which is not being delivered by HVE. With keywords such as, *automation, agile, C++* and *arkitektur* (architecture).

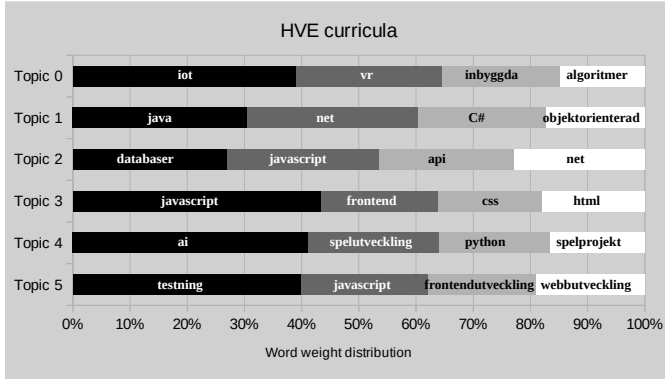
## VII. DISCUSSION

### A. Interpreting the results

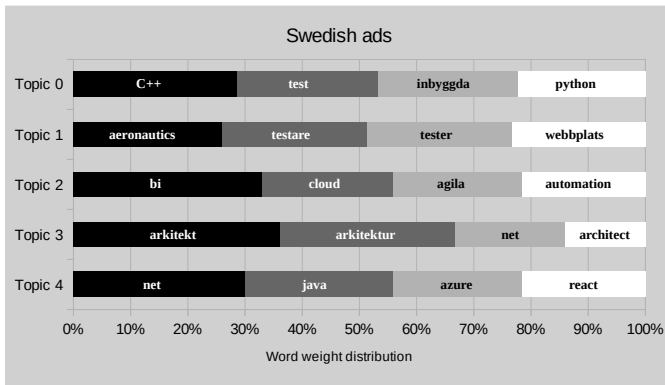
As mentioned in the results section, and can be seen in Figure 7a, many of the identified noise words among HVE have a very high document distribution. This is expected as all curricula are parsed from their corresponding provider's application and as these are based on a common template, the curricula have many words in common with each other. However, one word stands out from the rest, *medfinansiering* (co-financing), with a document distribution of 75%. This word gives us a hint about the number of HVE which are not only financed by the state. This information could be used to investigate differences between state-financed and industry-financed education, and whether they correspond differently to industry needs. Financing collaboration takes on many forms, such as lectures from consultants in the industry, study visits, learning in a work environment (LIA), and student industry meetings. In other words, even noise can provide us with valuable information. This was unexpected and should be noticed by anyone planning to conduct similar studies.

When it comes to the job ads there is a significant difference between the Swedish and English job ads Figs. 7b and 7c, in regards to the type of words identified as noise. The noise in the English job ads are more related to labour and recruitment, such as *development, data, knowledge*, than those in the Swedish job ads, for example *kunder, hos, söker*. Furthermore, the English job ads have a higher document distribution which tells us that these are more consistent. It is difficult to explain the cause of these differences, but is likely that English job ads in general are more consistently written compared with Swedish as there are often big international companies behind them. Another explanation could be the use of different stopword lists for the Swedish and the English language in the topic modelling process.

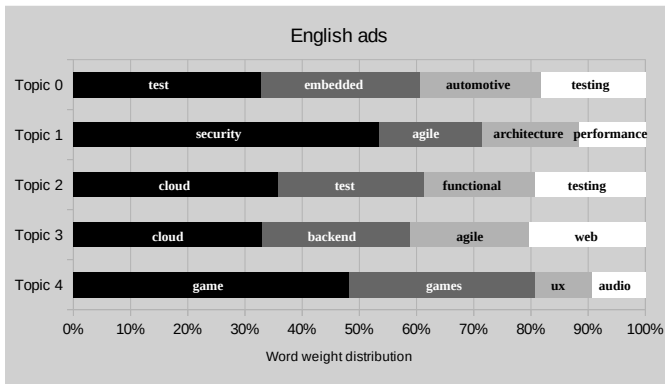
A number of categories were identified in the results section. Each of them corresponds to a topic in one or more of the datasets. The occurrence of each topic in its respective dataset can be seen in Tables IV–VI. *Topic 3*, identified as *architect*, in the Swedish jobs dataset stands out with its 151 occurrences, compared with the other job-related topics. But, its significance should not be overemphasized because its high number of occurrences is likely caused by one company,



(a) HVE curricula.



(b) Swedish job ads.



(c) English job ads.

Fig. 8: Identified topics and weight distribution of the top four keywords within each topic, where 100% is the sum of the word weights within each topic.

publishing many identical job ads for several regions. There are 533 Swedish job ads published by the company “Personal-expressen”. Three of its job ads are published 100-108 times, four of its jobs 50 times, and an additional three are published 10 or fewer times. All the topic’s keywords are present in 161 job ads, of which 150 are published by “Personalexpressen”.

This imposes a risk to the validity of this work and is brought up in *Threats to validity*. However, we still believe that the topic has some significance, with the words *architect* and *arkitekt*, which in the current context indicates a demand for *software architects*. Furthermore, the category *embedded* are present in both the Swedish and English datasets, which indicates a demand. The category *cloud* was found in both datasets but with very few occurrences in the Swedish dataset. Nevertheless, there is a clear demand for knowledge within the field. The category *web* has the most balanced number of occurrences over the job-related datasets and also indicates a demand for knowledge within the broad field of the web, including backend. Finally, the category *security*, which was only found among the English job ads, could indicate a higher demand for security-related knowledge among the international job ads, as there is no topic, nor keyword, in the Swedish dataset directly security related.

The study by Dobslaw et al. [6] came to the conclusion that the most sought-after skills by the industry were *Java*, *SQL* and *C#* at the time the study was conducted. It also measured the current trends and identified the two most uptrend technologies being *Docker* and *Kubernetes*, which both are technologies used for the packaging and distribution of software for the cloud. It is possible the skills demands of the industry have changed during the year since the study by Dobslaw et al., but it is interesting how well the identified trend corresponds with the results of this paper.

Furthermore, HVE curricula are mostly web-focused but also offer knowledge within object-oriented programming. This also reflects our experience from what we have seen while developing VELMA. *Topic 4* in of HVE dataset, which was also mentioned in the results section, contains the keywords *ai*, *spelutveckling* (game development), *python*, *spelprojekt* (game project). While its number of occurrences is very low, the combination of keywords provides an indication that game development, artificial intelligence (AI) and the programming language python, tend to occur together in curricula.

By looking at the results from **RQ2** and **RQ3** it appears that the technologies in demand by the industry today are cloud, embedded, security and architectural related. Whereas in the HVE dataset, there is no topic with any document occurrence covering the aforementioned technologies. With that said, one term from *topic 0* of the Swedish dataset, *inbyggda*, corresponding to *embedded*, is found in *topic 0* in the HVE dataset. By observing the latter topic we can observe a foreign term, unrelated to the rest; *vr*. This indicates a badly generated topic, and by looking at the rest of the terms we can see terms that are often related to embedded development; *iot*, *inbyggda* (embedded) and *algoritmer* (algorithms). Because of how these terms related to embedded were grouped, we

can say for certain that at least some HVE cover these technologies. As such, we can identify a flaw in the topic modelling process which indicates the results should not be taken at face value.

Despite the identified flaws, based on our findings we can recommend the following to HVE providers and to the industry.

- HVE should shift focus from web development to cloud-, embedded-, and security-related technologies together with architectural disciplines.
- Job sites should not allow companies to publish more than one job ad per vacancy.
- Job sites should implement a template in which the recruiter must fill out wanted skills in a structured manner.

## B. VELMA

VELMA in its current state is focused on fetching only HVE curricula and Swedish job ads within software development and programming. However, in order to change industry domains, while remaining within HVE and job ads, the parameters used in the API requests need to be changed in accordance with the SUN- and SSKY-codes. Furthermore, by implementing calls to new APIs, the scope of VELMA can be further extended. For example, HVE and general higher education could be compared by fetching university curricula. Industry trends could also be explored by fetching historical job ads. Moreover, because of the dynamic nature of topic modelling, skills at an abstract level such as soft skills could be explored by modifying the stopwords list, aligning with what was done by Andersson et al. [7], [8].

## C. NLP

The use of unsupervised machine learning, as in the case of LDA topic modelling, typically introduces some uncertainties. In the case of this study, we encountered several obstacles while developing and refining the topic modeller. First and foremost, we found the *issue with noise* to be quite large, mainly because of the verbose text of job ads. As can be seen in the number of iterations needed to “clean up” the noise in the respective dataset in Table II.

Moreover, *choosing the “correct” parameters* for the modelling process proved to be hard, since what is the “best” in terms of evaluation measures might not always be the case from a human-readable perspective. More specifically, two evaluation methods were explored during the study; (i) coherence and (ii) log-likelihood. Where we found the best parameters of the respective evaluation process to be contradictory, that said, the parameters for the best coherence score would equate to the parameters of the worse log-likelihood. In the end, the modeller was run on the respective evaluation method’s best and worst parameters and found the log-likelihood to produce the most reasonable topics. The issue with optimal parameters was also brought up by Gurcan and Kose [13], where they also cover the issue with multilingual job ads. This was also noted by us during the development process, not only regarding the large quantities of ads written

in English but also where a small number of job ads composed of Norwegian and Danish languages outside of Swedish were identified. However, because of their minuscule number, this turned out to not impact the modelling process.

## D. Threats to validity

There are several possible sources of threats to the validity. One of them is that the noise has been manually selected by us, and the risk of bias. Another possible source of threat to validity is the stopword lists for Swedish and English, which could be of low quality. Furthermore, the results from VELMA can be difficult to interpret. There is the risk of bias when we make the interpretations. The difficulty consists of two parts; the first is identifying each topic found by the LDA modeller. In many cases, this is quite easy, but sometimes two categories are mixed in one topic, or two topics have keywords that seem to be of the same category. The second part is grouping the identified topics between the datasets, in order to be able to draw conclusions. These threats are difficult to fully mitigate, but we have taken some steps in order to minimise those related to the interpretation of the results, such as *internally discussing* and *conducting searches on the Internet* before making any decisions regarding the categorisation of data.

There is also a risk of unbalanced input of job ads to the LDA modeller. We have noticed one company publishing the same job ad multiple times in different regions while it is not likely that the job exists in all these regions. Such data could be manually removed before feeding it to the LDA modeller, but that also produces a risk of bias; what if the company really is offering the exact same job in all these regions? Unfortunately, this threat was discovered at a late stage during the study, and could not be mitigated due to time constraints. It is hard to judge how the duplicated job ads have impacted on our findings. However, by identifying the problem, important knowledge can be passed on to future works within the area where this issue can be mitigated.

## E. Societal and Ethical aspects

One prevalent ethical aspect in our research was related to the use of the API owned by MYH. While MYH’s website supplies the public with all kinds of documents, such as the HVE programme applications that VELMA use, it is not designed to let anyone download several documents at once. VELMA, however, is built to download many documents in a row. This could cause an abnormal load on their server. The existence of the API is not public, so it might be questionable if it is allowed to be used. Moreover, when we contacted MYH’s IT department regarding the API, we were told no such API exists. Based on the very short period of time that VELMA possibly could cause an extensive server load, in relation to the benefit of getting the data, we decided to go ahead with the implementation and use of VELMA when producing the results for this thesis. Moreover, a societal aspect is that one company, “Personalexpressen”, discussed in *threats to validity*, which imposes a negative impact on the validity of the results by its many identical job ads. Such

behaviour poses risks to the validity of not only this study but to any study using quantitative data based on job ads.

Finally, this study and VELMA provide means for the HVE providers to improve their curricula to better meet the industry's demand, as well as for anyone planning to start a new HVE programme. Furthermore, recruiters of the industry can use VELMA to better understand the competencies of the graduates. In any case, the outcome is positive for society as a whole.

## VIII. CONCLUSIONS

This study collected HVE curricula and job ads and applied the NLP process, LDA, together with document frequency analysis on the data to investigate the balance of the Swedish software industry and HVE. The results show that cloud-, embedded-, and security-related technologies together with software architects are in demand which is not being delivered by HVE. Moreover, findings show that a large focus of HVE programmes is on web-related technologies and that HVE covers popular object-oriented programming languages. Furthermore, by investigating the noise in the HVE curricula and job ads, one unexpected finding regarding the co-financing of HVE programmes was observed. Indicating an increased industry collaboration in the majority of HVE programmes via lectures from consultants and student visits, among others. Additionally, findings in the job ads dataset show that ads written in Swedish are noisier than those written in English. Future work should consider improving the topic modelling process, either via extending on LDA or exploring other models such as HLTMs. An improvement of the data pre-processing should also be considered for the reason that companies publish duplicate job ads with differentiating IDs which can impact the findings. Moreover, VELMA, or the stopwords list generated for the noise, could function as a baseline for further expansion on improving the modelling process. Additionally, the addition of an API for fetching curricula from general higher education could increase the scope of VELMA and make it an important tool for universities as well.

## REFERENCES

- [1] S. J. Patrick Joyce, "Almeas tjänsteindikator," <https://www.almega.se/app/uploads/2022/03/tjansteindikatorn-kvartal-ett-2022-slutversion.pdf>, 2022 (accessed March 02, 2023).
- [2] SCB, "Arbetskraftsbarometern 2021 – vilka utbildningar ger jobb?" [https://www.scb.se/contentassets/2523aa42021a40e38675e630a327b706/uf0505\\_2021a01\\_am78br2201.pdf](https://www.scb.se/contentassets/2523aa42021a40e38675e630a327b706/uf0505_2021a01_am78br2201.pdf), 2021 (accessed February 16, 2023).
- [3] D. Spetskompetens, "Flöden av digital spetskompetens," <https://digitalspetskompetens.se/rapporter/floden-av-digital-spetskompetens/>, 2022 (accessed February 16, 2022).
- [4] S. Näringsliv, "Växande rekryteringshinder ett allt större problem," <https://www.svensktnaringsliv.se/sakomraden/utbildning/vaxande-rekryteringshinder-ett-allt-storre-problem-rekryteringsen-1182984.html/>, 2021 (accessed February 16, 2023).
- [5] V. Bodell, "Svenska datautbildningsrelevans för mjukvaruutveckling inom industri," 2020. [Online]. Available: <http://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A1417121&dsid=-474>
- [6] F. Dobslaw, K. Angelin, L.-M. Öberg, and A. Ahmad, "The gap between higher education and the software industry – a case study on technology differences," in *Proceedings of the 5th European Conference of Software Engineering Education*, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2303.15597>
- [7] M. Andersson, A. Kusotogullari, and J. Wernberg, "Software development and innovation: Exploring the software shift in innovation in Swedish firms," *Technological Forecasting and Social Change*, vol. 167, p. 120695, 2021. [Online]. Available: <https://doi.org/10.1016/j.techfore.2021.120695>
- [8] J. W. Martin Andersson, "Den programmeringsbara ekonomin," <https://www.ifn.se/publikationer/rapporter/2011-2020/2020/2020-15>, 2020 (accessed February 21, 2023).
- [9] S. Sweden, "Trender och prognoser 2020," <https://www.scb.se/publikation/39441>, 2021 (accessed March 02, 2023).
- [10] P. F. Åsa Zetterberg, "The it competence shortage," <https://www.almega.se/app/uploads/sites/2/2020/12/ittelekomforetagen-it-kompetensbristen-2020-eng-online-version-2.pdf>, 2020 (accessed March 02, 2023).
- [11] A. Schofield and D. Mimno, "Comparing apples to apple: The effects of stemmers on topic models," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 287–300, 2016. [Online]. Available: [https://doi.org/10.1162/tac1\\_a\\_00099](https://doi.org/10.1162/tac1_a_00099)
- [12] M. Steyvers and T. Griffiths, "Probabilistic topic models," in *Handbook of latent semantic analysis*. Psychology Press, 2007, pp. 439–460. [Online]. Available: <https://cocosci.princeton.edu/tom/papers/SteyversGriffiths.pdf>
- [13] F. Gurcan and C. Kose, "Analysis of software engineering industry needs and trends: Implications for education," *International Journal of Engineering Education*, vol. 33, no. 4, pp. 1361–1368, 2017. [Online]. Available: [https://www.researchgate.net/publication/318582283\\_Analysis\\_of\\_software\\_engineering\\_industry\\_needs\\_and\\_trends\\_Implications\\_for\\_education](https://www.researchgate.net/publication/318582283_Analysis_of_software_engineering_industry_needs_and_trends_Implications_for_education)
- [14] A. K. McCallum, "Mallet: A machine learning for language toolkit," 2002, <http://mallet.cs.umass.edu>. [Online]. Available: <http://mallet.cs.umass.edu>
- [15] S. Lee, J. Song, and Y. Kim, "An empirical comparison of four text mining methods," *Journal of Computer Information Systems*, vol. 51, no. 1, pp. 1–10, 2010. [Online]. Available: <https://www.tandfonline.com/doi/abs/10.1080/08874417.2010.11645444>
- [16] P. Chen, N. L. Zhang, T. Liu, L. K. Poon, Z. Chen, and F. Khawar, "Latent tree models for hierarchical topic detection," *Artificial Intelligence*, vol. 250, pp. 105–124, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0004370217300735>
- [17] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003. [Online]. Available: <https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf>
- [18] P. Kherwa and P. Bansal, "Topic modeling: A comprehensive review," *ICST Transactions on Scalable Information Systems*, vol. 7, p. 159623, 07 2018. [Online]. Available: <https://eudl.eu/doi/10.4108/eai.13-7-2018.159623>
- [19] M. Gethers and D. Poshyvanik, "Using relational topic models to capture coupling among classes in object-oriented software systems," in *2010 IEEE International Conference on Software Maintenance*, 2010, pp. 1–10. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/5609687>
- [20] X. Yang, D. Lo, L. Li, X. Xia, T. F. Bissyandé, and J. Klein, "Characterizing malicious android apps by mining topic-specific data flow signatures," *Information and Software Technology*, vol. 90, pp. 27–39, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S095058491730366X>
- [21] K. Bastani, H. Namavari, and J. Shaffer, "Latent dirichlet allocation (lda) for topic modeling of the cfpb consumer complaints," *Expert Systems with Applications*, vol. 127, pp. 256–271, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S095741741930154X>
- [22] U. Chauhan and A. Shah, "Topic modeling using latent dirichlet allocation: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 7, pp. 1–35, 2021. [Online]. Available: <https://dl.acm.org/doi/abs/10.1145/3462478>
- [23] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno, "Evaluation methods for topic models," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 1105–1112. [Online]. Available: <https://dl.acm.org/doi/abs/10.1145/1553374.1553515>

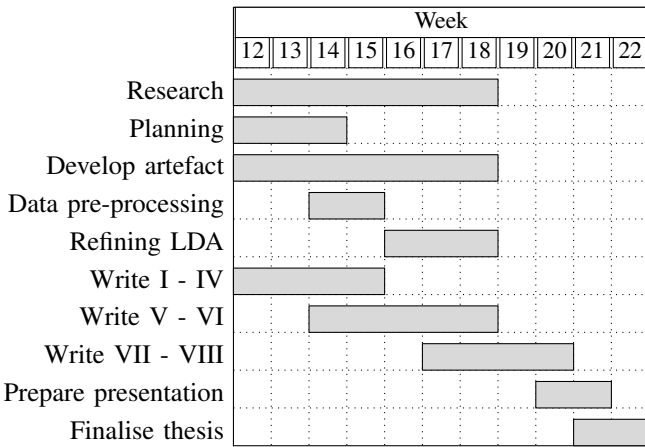


Fig. 9: Initial time plan.

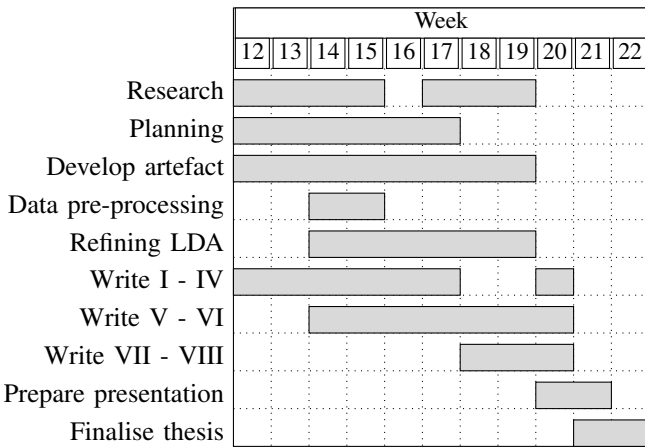


Fig. 10: Final time plan.

#### APPENDIX 1: TIME PLAN

See the respective Figs. 9 and 10 for the initial and final time plan for the study.

#### APPENDIX 2: CONTRIBUTIONS

Overall, the authors contributed equally, but in the listed areas below respective author devoted more time.

- Emil Bäckstrand

- Thesis

- \* Background (Topic model)
    - \* Related Work
    - \* Research Methodology (Data pre-processing)
    - \* Results (RQ1, RQ3, RQ4)
    - \* Discussion (NLP)
    - \* Conclusion

- VELMA

- \* Job ads language filter
    - \* PDF parser
    - \* Corpora creator
    - \* Topic modeller

- Rasmus Djupedal

- Thesis

- \* Background (Higher Vocational Education)
    - \* Research Methodology (Artefact creation, Data collecting)
    - \* Results (RQ2)
    - \* Discussion (Interpreting the results, Threats to validity, Societal and Ethical aspects)

- VELMA

- \* GUI
    - \* API parsers
    - \* Keyword analyser