**Topic Modeling & NLP: STEM Apprenticeships**

By Alphonse Simon

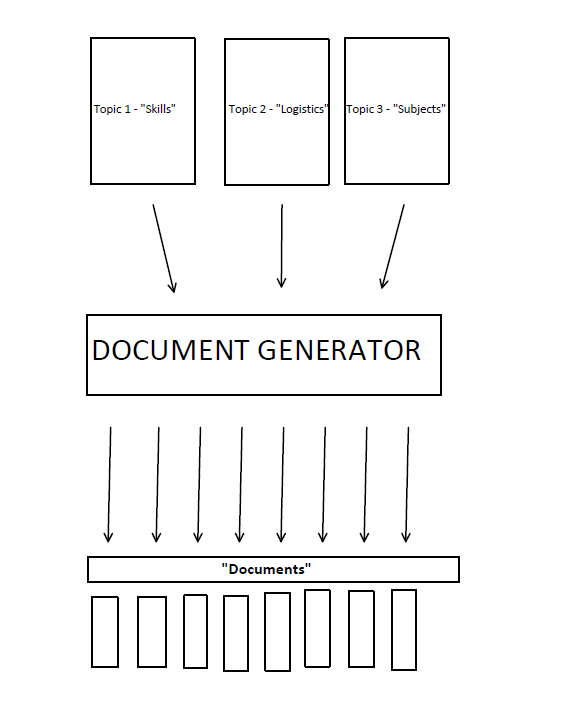
1. ***Goal***

The goal of this project is to analyze the difference between the skills gained in the On the Job Training portion of STEM Apprenticeships and the skills gained from a similar academic program at the Community College level (usually resulting in an A.S).

1. **Method**

The preliminary method used in the project is a form of Natural Language Processing called Latent Dirichlet Allocation Topic Modeling. “Latent” is the key word, as the model seeks to uncover abstract “hidden” topics that describe documents.

The preliminary model used recognizes a user-specified number of topics. The model assumes topics are made up of “words” which are given weights to signify how meaningful they are in describing the model. The model assumes are there are documents, which in this case are our work process schedules and our course descriptions. The model (incorrectly, but it’s important to know this) assumes that these documents are created in a random process. Prior to the document creation, each document is given a series of weights attached to each topic which signifies how much of the topic that document is comprised of. So, using our figure below, document #1 could be .5\*topic\_1(“skills”) and .5\*topic\_2(“logistics”). Once this weight is set, the words from the topics are randomly selected (the higher the weight the more likely it is to be selected) and they create the document.



1. **Data**

We have two different datasets being trained: OJT from Work Process Schedules & Community College course requirements for IT related fields. Community College course requirements are pulled from a random group of Community Colleges (around 40 in total) who were listed in the DOL’s RACC. The requirements for each of the programs were grabbed and the course descriptions were compiled into one document for each program.

1. **Results**

*Work Process - OJT*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Topic 1 |  | Topic 2 |  | Topic 3 |  | Topic 4 |  |
| **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** |
| 0.002 | "manag" | 0.028 | "databas" | 0.017 | "configur" | 0.036 | "organ" |
| 0.002 | "organ" | 0.019 | "secur" | 0.017 | "secur" | 0.021 | "success" |
| 0.002 | "secur" | 0.018 | "identifi" | 0.017 | "manag" | 0.019 | "practic" |
| 0.002 | "develop" | 0.016 | "document" | 0.014 | "develop" | 0.015 | "develop" |
| 0.001 | "servic" | 0.016 | "requir" | 0.014 | "network" | 0.015 | "apprentic" |
| 0.001 | "network" | 0.014 | "configur" | 0.010 | "servic" | 0.015 | "procedur" |
| 0.001 | "configur" | 0.013 | "manag" | 0.010 | "organ" | 0.014 | "network" |
| 0.001 | "work" | 0.013 | "data" | 0.009 | "plan" | 0.014 | "present" |
| 0.001 | "instal" | 0.013 | "window" | 0.009 | "process" | 0.013 | "servic" |
| 0.001 | "perform" | 0.013 | "plan" | 0.009 | "server" | 0.013 | "configur" |

Coherence score = 0.28597

*Community College by Program*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Topic 1 |  | Topic 2 |  | Topic 3 |  | Topic 4 |  |
| **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** |
| 0.016 | "design" | 0.015 | "secur" | 0.022 | "develop" | 0.018 | "program" |
| 0.013 | "certif" | 0.014 | "oper" | 0.016 | "basic" | 0.014 | "read" |
| 0.012 | "develop" | 0.014 | "provid" | 0.015 | "applic" | 0.013 | "prerequisit" |
| 0.012 | "prepar" | 0.012 | "manag" | 0.014 | "secur" | 0.012 | "write" |
| 0.011 | "softwar" | 0.012 | "configur" | 0.013 | "design" | 0.011 | "applic" |
| 0.011 | "manag" | 0.012 | "certif" | 0.013 | "provid" | 0.010 | "pend" |
| 0.011 | "oper" | 0.011 | "prepar" | 0.013 | "program" | 0.009 | "requir" |
| 0.010 | "program" | 0.011 | "design" | 0.013 | "softwar" | 0.009 | "account" |
| 0.010 | "instal" | 0.010 | "technolog" | 0.012 | "technolog" | 0.008 | "area" |
| 0.009 | "basic" | 0.010 | "window" | 0.011 | "oper" | 0.008 | "experi" |

Coherence score = 0.37106

[View Notebook with PyLDAvis](C://Users/alpho/Downloads/PyLDAVis.html)

1. Next Steps/Discussion

I don’t think the results make much sense for the project’s goal. The hope was that there would be a noticeable “skills” topic. It seems in retrospect (maybe?) unlikely that a skills topic would be one of the topics generated, more so for data reasons: if community colleges offer similar skills as each other, the skills won’t be captured as a unique topic, likewise for the work process schedules. Or it could be a methodological issue: skills aren’t used at a high enough frequency in these documents and the current method does not capture their use. For these I propose a few solutions:

If the issue is a ***data*** issue, then we could do the following:

1. Train the two datasets together, with the hope that apprenticeship skills and community college skills is at least somewhat captured in a topic
   1. I tried this, and it didn’t work too well, but the # of community colleges far outnumber the #of schedules. So, for this we could just
2. Experiment with the number of topics (right now I picked 4 arbitrarily) [insert coherence score and perplexity score – with a primer on how they’re calculated]
3. Include non-IT related fields (expanding the number of topics to (# of fields\*3) so that the topics can capture the field, skills and some other aspect)
4. Expand “stop w0rd” list.
   1. Essentially, I’ve written code to remove meaningless words
5. Splitting up the documents
   1. Training on the individual classes as documents, as opposed to the entire programs, providers a similar coherence score
   2. This would mean splitting up the

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Topic 1 |  | Topic 2 |  | Topic 3 |  | Topic 4 |  |
| **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** | **Weight** | **Word** |
| 0.002 | "manag" | 0.028 | "databas" | 0.017 | "configur" | 0.036 | "organ" |
| 0.002 | "organ" | 0.019 | "secur" | 0.017 | "secur" | 0.021 | "success" |
| 0.002 | "secur" | 0.018 | "identifi" | 0.017 | "manag" | 0.019 | "practic" |
| 0.002 | "develop" | 0.016 | "document" | 0.014 | "develop" | 0.015 | "develop" |
| 0.001 | "servic" | 0.016 | "requir" | 0.014 | "network" | 0.015 | "apprentic" |
| 0.001 | "network" | 0.014 | "configur" | 0.010 | "servic" | 0.015 | "procedur" |
| 0.001 | "configur" | 0.013 | "manag" | 0.010 | "organ" | 0.014 | "network" |
| 0.001 | "work" | 0.013 | "data" | 0.009 | "plan" | 0.014 | "present" |
| 0.001 | "instal" | 0.013 | "window" | 0.009 | "process" | 0.013 | "servic" |
| 0.001 | "perform" | 0.013 | "plan" | 0.009 | "server" | 0.013 | "configur" |

Coherence Score = 0.374408

[View PyLDAvis for this.](C://Users/alpho/Downloads/PyLDAVis-2.html)

If this is a ***methodological*** issue, then we could do the following:

1. Utilize a skills extractor like [Skills-ML](https://github.com/workforce-data-initiative/skills-ml/blob/master/Skills-ML%20Tour.ipynb).
   1. I attempted this and spent an embarrassingly long time trying to get it to work on my computer (to no avail ☹)
   2. Core idea is that Skills-ML has pre-trained data off ONET data. They use word-embeddings which give words vectors that describe their meaning. You can use these word-embeddings to test the data we have and see which skills are represented well. You can then run all sorts of analysis on this data.
2. [MALLET](https://radimrehurek.com/gensim/models/ldamallet.html)
   1. Another LDA topic modeling program that provides better topics. I again, would’ve done this but had technical issues.
3. [Semi-Supervised training](https://medium.freecodecamp.org/how-we-changed-unsupervised-lda-to-semi-supervised-guidedlda-e36a95f3a164)
   1. This method allows you to train a model to teach it how to identify skills.
4. [LDA2Vec (Word Embeddings with Topic Modeling)](https://www.datacamp.com/community/tutorials/lda2vec-topic-model)
   1. This method adds semantical meaning to words so LDA models will consider meaning of words when creating topics.
   2. I would’ve done this because it seems obvious – but an intense amount of data is required (so I guess this fits into both the data and methodological concern).