LinkedIn Recruitment Systems through Social Graphs

Jordi Beernink , Brian Westerweel, *Student, IEEE*

***Abstract*—The purpose of this paper is to review different recommendation models that uses user and skill information to recommend either possible candidates or relevant skills. This paper focuses on the principles of frequency of items, user-item matrices and word-vector similarities to make these recommendations. In the case of possible candidates, social network analysis and graph theory will be implemented to see how the recruiter and the candidate are possibly connected. This is done through user features such as shared companies, educational background, skillset and location. This gives the recruiter more insight in the model's recommendations of possible candidates based on the required skillsets. The recommendations of the models will be rated and will be analyzed to see how well they performed.**

I. Introduction

A common problem for companies is finding the right persons to hire. In the past few decades application processes have started to go digital. This results that applicants are nowadays looking on various sites for vacancies and companies are using these sites to find people for the available jobs. Examples of these are Monsterboard, Vacaturebank, Indeed and many others. These websites can help companies find persons to hire based on properties such as educations, experiences and ambitions. However, it is still possible to improve on finding the right person. Because companies do not only want persons to have the required skills or experience, but they also want employees that fit the philosophy of the company[1].

Besides these helpful websites companies are also looking for new employees on the social network application: LinkedIn. Because LinkedIn is able to create relations between different persons. These relations are made when people meet each other on an university, company or conference. These relations can be created due to the persons having similar skills or education, interests and mentalities[2].

In this paper we incorporate these relations, also known as *social networks*, in a recommender system to give recruiters the possibility to choose how closely related to the company they want an applicant to be. This would create the option to search for new employees that have similar mentalities or have very different ones so the company can acquire some fresh ideas.

S4850750/S4802594
24th June 2018, George Kachergis
email: J.A.Beernink@student.ru.nl, B.Westerweel@student.ru.nl

II. Background

Social network analysis is a method for investigating social structures between persons[3]. Social network is the structure, which shows the relations between individuals and organization. It indicates the ways in which they are connected through various social familiarities ranging from casual acquaintances to close familiar bonds[4]. These social structures can be viewed from the traditional individualistic social theory and data analysis, which considers individual actors making choices without taking the actions, preferences and behavior of other users into consideration. This approach omits the social interactions and relationships of the actor. The primary concern in this is the property of the actor itself[5].

In social network analysis, however, the relationships form the primary concern while the properties of the actor become secondary. The relations between persons will be the main focus of this project. While the characteristics of the individual will be used to fully understand how these social interactions came to be and how the user can be represented[5]. For this project the following definition will be used as the starting point for the recommender systems[6].

"Most broadly, social network analysis (1) conceptualizes social structure as a network with ties connecting members and channelling resources, (2) focuses on the characteristics of ties rather than on the characteristics of the individual members, and (3) views communities as 'personal communities', that is, as networks of individual relations that people foster, maintain, and use in the course of their daily lives[5]"

The last aspect of social network analysis that will be discussed are the two forms that can be distinguished: the ego network analysis and the global network analysis. In 'ego' studies the network of one person is analyzed. An example of this is based on the description of White of the research network centred on Eugene Garfield[7]. In 'ego' studies the network of one person is analyzed, while in global network analysis one tries to find all relations between the participants of the network[7]. The 'ego' method will be used as a way to analyze the network of the recruiter, while the global method will be used to find clusters of persons based on their background.

A. Graph Theory

Social networks can be represented as an undirected graph G(V,E) where V is the set of vertices representing users and E is the set of edges representing relationships[4]. Graphs are used to represent communication networks or in the case of this project, social networks. The analysis of these social networks properties and user interactions can help with understanding and improving networking solutions[8]. In the case of relationships , these can modelled in the following way.

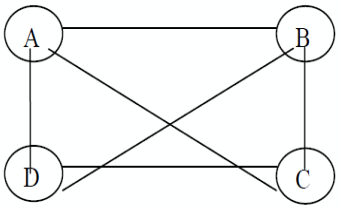

Fig. 1. Connectivity among four vertices

In this figure A, B, C, D are considered friends and each of them have interactions with each other. This figure is based on direct paths in which all nodes know each other directly. A variance on this is when a community is introduced in which both A and B are connected through it and where a direct edge is present between A and B[4].

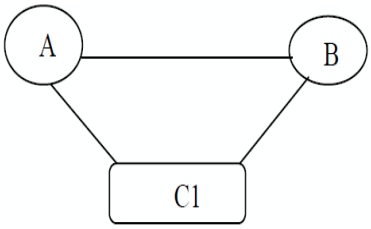

Fig. 2. Common community between two vertices

A and B are two connected friends, while C1 is a community in which they both participate. Hence, A and B are connected through two different paths. One is the direct connection and another is a connection through community C1. Based on this a conclusion can be made that the connection between A and B are strong, as they belong to the same community and share a direct path. However, an example when this direct path is not present between the nodes can be seen in the next figure[4].

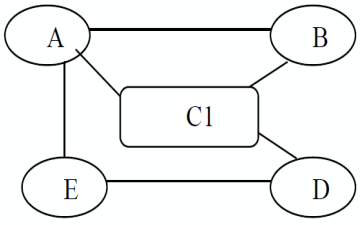

Fig. 3. Common community between multiple vertices with weak and strong ties

Here A and B are again strongly connected through their direct path. Similarly, the connection between A D and B D is weak due to only having a indirect path through community C1. The connections between A E and E D can be considered weak based on interpretation due to only having a direct edge between the nodes. Lastly, B E are connected through A, i.e. A is a common node between B E. With this E can contact B through A[4].

These direct and indirect connections and the communities will be used for the recommender systems. Because the recruiters can check in their own social network where relevant persons are positioned, while also using their own connections to contact persons outside their social network.

B. Recommender Systems

In current society recommender systems are not anything new. Most of them are found in web shops and services like Netflix that recommend articles to users based on different aspects such as behavior, preferences and similar articles [9]. Recommender systems are defined as a decision making strategy for users under complex information environments [10]. An example of this is to help users search through records of knowledge and articles based on the user's interest and preferences [10]. Recommender systems handle the problem information overload that users normally encounter by providing them with personalized content. To support individuals with this problem a lot previous research has already been performed [10][11]. This resulted in different methods and models to help create recommendations for individuals. For this project the following recommendation models will be implemented.

- Popularity
- Content-Based Filtering
- Collaborative Filtering
- Hybrid of Content and Collaborative

All these methods share the same phases to generate recommendations. The first steps consist of collecting information to generate an user profile for their prediction tasks. These can include user's attribute, behaviors and/or previously interacted items of that specific user. The success of the agent depends largely on its ability to represent user's current interest. If the interests of the model and the agent do not align, the agent will not be able to consistently make correct recommendations [10].

To achieve more accurate recommendations the model needs users feedback on the items to improve itself. The higher the quantity of ratings and interactions provided by the user the more likely the accuracy of the model will improve or worsen. The following figure shows the process from collecting, making recommendations and learning from the feedback. By continuously getting feedback from the user. The model can continuously learn from the user based on their actions[10].

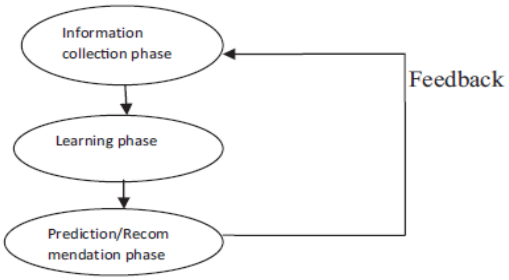

Fig. 4. Recommendation Process

This principle of continuously sending feedback to the model through user interaction is applicable to all recommendation systems used in this project. However, because different models can be (un)suitable for different recommendations[10]. The following models will be implemented for this project.

B.1 Popularity Model

The model does not focus on the user's preferences or actions and only looks at the popularity within the item list. More popular items are more likely to be recommended, while items that are less popular will appear less likely or not at all. This model is not based on theoretical grounds due to the fact that the it only looks at the frequency of interactions of the item.

B.2 Content-Based Model

The technique emphasizes more on the analysis of the attributes of items in order to generate predictions[12]. Examples of these are web pages, publications and news. Items with similar features are grouped together and will be recommended to the user based on their previous interacted items[12]. The content-based model can use different types of processing and techniques to generate these recommendations. Examples of these are vector space models, TF/IDF, Decision Trees which are used to compute the similarities between the different documents within a corpus[10]. The advantages of this model is that no domain knowledge is needed to perform the analysis. Because the recommender is able to spot the similarities within the features without needing any prior influences. The drawbacks of this method is that the quality of the recommender agent is dependent on the size of the data set and cannot correctly make predictions for *cold-start* users and items[13].

B.3 Collaborative Filtering Model

This technique generates an user-item matrix based on the previously interacted items of the users. It then matches users with similar item interactions by calculating the similarities between their profiles to generate the recommendations[13]. Users that have a high similarity will form a cluster and will be used as reference by the model for new users. By recommending non-interacted items from a list of interactions from that user cluster to new users that appear to be similar to that specific cluster. The advantages of this model are similar to the Content-based model. However, Collaborative filtering is also able to recommend niche articles that the Content-based will miss. Because the Content-based model will only check items with similar features, while Collaborative checks the user-item interactions and is able to spot niche user-item interactions. As a result, the drawback of the model is that "gray sheeps" can be created. Because there is a possibility that a user's interaction does not appear to be similar or match to existing user clusters [14][15].

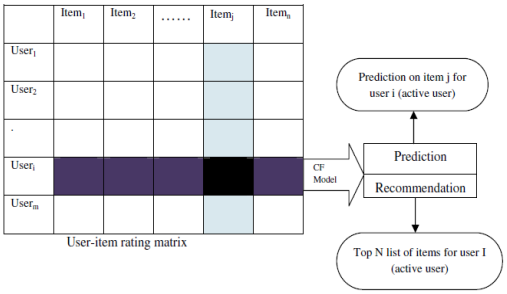

Fig. 5. User-Item Interaction Matrix

C. Implementation

The difference between previously written work and research is that the recommendation made by these models will be evaluated by the authors. In this project a sample of the recommendations will be rated manually to see if the accuracy scores with the model coincide with the manual rating.

Another difference is the introduction of the popularity model. This model will serve as the baseline for the other models and will represent the *human recruiter*. With this the more popular users and items will represent the more requested searches, while unknown skills and less easy to find persons will appear less.

The last difference is that both users and items are being recommended by the same models. By doing this the different feature constructions of both items and users can be used to see which model performs better with each construction.

III. Experiment

The following section describes the different approaches in this project and the used dataset will be explained in detail. The section concludes with the explanation of the evaluation methods.

A. Approach

The approach of this experiment can be split into two stages. In the first stage a sample of the users of the dataset will be used to recommend skills based on their user profile. While, the second stage will consist of finding people with specific key skills and see if they are present in the recruiter's social network. This will be done through a

small case study in which a random user will be grabbed from the dataset and needs to find persons with specific key skills.

A.1 Recommending skills to people

This approach focuses on recommending skills to people from a sample of random users within the dataset. By using this approach the users profiles can be analyzed to see if the recommended skills fit with their user profile and could be relevant from them to learn.

A.2 Recommending people based on skills

This approach focuses on finding individuals based on key skills that the recruiter needs to fulfill for their job position. For this the *recruiter* will get recommendations of the models based on specific required key skills. Afterwards the recommended persons will be checked to see if they are present in the recruiter's social network either directly, indirectly or not at all.

B. Data set

The data set used for this project is a subset of a LinkedIn scrape. It was difficult to scrape the data directly from LinkedIn because of LinkedIn's privacy policy which only allows scraping when LinkedIn has given consent. Because of time constraints this was not possible for this project. The available scrape consisted of first and last names of users, industry, positions, education background and acquired skills. After preprocessing the data set consisted of 7234 unique users and 19329 unique skills. Because of the relative high frequency of skills the training and test sets will be decreased based on the amount of unique user-item interaction.

For this project the data set will be split into users with at least three and at least ten item interactions. With this the cold-start problem will be emphasized in the case of low user-item interactions, while the high user-item interactions is assumed to create more distinct user clusters.

C. Evaluation

The evaluation will be done in three parts. The first part will consists of applying precision, recall and F1-Score to the different models on the low and high user-item interactions. The metrics will be applied by the first 5 and 10 recommendations. The formula used for these metrics can be found in *Appendix I: Mathematical Formulas.*

The second evaluation will consists of manually rating the top 5 skill recommendations for 10 users. These manually ratings will consist of rating the relevancy from 1 (Not Applicable) to 5 (Applicable). The recommendation list will be made available in the github repository with the corresponding rating scores. An average will then be calculated over these scores resulting in a performance score of applicability of each model.

The third evaluation will consist of manually rating the top 5 recommended users based on 10 skills. In this the recommended persons will be checked based on their position in the network and how applicable these persons are. The applicability rating will be done in the same approach as the recommended skills approach, while the positioning will be based on the likelihood users know each other. Due to the data set being a subset of an entire scrape the connections between users are not present, only the amount of connections are known. Because of this all the user features were concatenated and afterwards TFIDF vectorized into a similarity matrix. The linear kernel was applied to this method to classify the different users.

Afterwards a threshold of 0.3 similarity was introduced to check if a relationship was possible between the recruiter and the possible candidate, where above 0.3 can be considered a relationship and under 0.3 the relationship is not possible. This threshold was chosen because of the messy data set and to lower the possibility that people with only similar job positions were joined together.

The manual rating will be done through a case study where a user from the data set will be used as a recruiter to find persons with specific key skills. If the person can be reached through a direct or indirect path the person can be considered a candidate and a distance score of either 5 or 3 will be put down at that person. If the person is outside the range. The person is very unlikely to be noticed and their distance score will be a 1. Because there is no direct or indirect path available to contact to reach that applicant.

The features of the users that will be vectorized for the similarity matrix are:

- Educational background: School, Name, Start and End Date
- Position: Title, Company, Start and End Data
- Location and Country
- First name

IV. Results

In this section the results of the different models will be presented.

A. Recommending skills to people

The metric scores are marginally different when it comes to either people with few or many skills. In this situation both models, purely based on precision, recall and F1-Score, can be used to recommend skills to people.

The Collaborative filtering model actually performs worse than originally expected, because it was expected to outperform the Content model due to the fact that it is able to find niche items and incorporates a user-item matrix. However, the Collaborative model performs 15 and 30 percent worse when compared to Content and the Hybrid model respectively.

The Content model performs really well, while the Hybrid model actually is able to reach a perfect precision by adding the Collaborativerecommendations to the Content model.

Model	Precision @5	Precision @10
3 Skills Popularity	39.0	52.2
3 Skills Collab	55.6	67.7
3 Skills Content	73.4	84.7
3 Skills Hybrid	99.7	100
10 Skills Popularity	38.4	51.7
10 Skills Collab	56.0	67.8
10 Skills Content	67.3	81.2
10 Skills Hybrid	99.7	100

TABLE I
Recommending Skills to Individuals - Precision Scores

As can be seen the recall scores are about the same as the precision scores. However, the Content model outperforms the Hybrid model, this could be explained due to the fact that the Hybrid model gets influenced a lot by the lower scores of the Collaborative model.

Model	Recall @5	Recall @10
3 Skills Popularity	29.6	39.6
3 Skills Collab	38.2	46.4
3 Skills Content	68.0	78.4
3 Skills Hybrid	58.8	59.0
10 Skills Popularity	29.6	39.9
10 Skills Collab	38.6	46.8
10 Skills Content	61.3	74.0
10 Skills Hybrid	58.4	58.5

TABLE II
Recommending Skills to Individuals - Recall scores

F1-Score is calculated by using precision and recall. As a result the scores are as expected. Where the best model according to the predictions are Content and the Hybrid model. Also, an interesting development can be spotted at the 3 and 10 Skills Hybrid score difference between the hits. There is only a small increase of .2 and .3 respectively. This is due to the recall influence of the Hybrid that shows the same behavior as the F1-Score.

Model	F1-Score @5	F1-Score @10
3 Skills Popularity	33.7	45.0
3 Skills Collab	45.3	55.1
3 Skills Content	70.6	81.5
3 Skills Hybrid	74.0	74.2
10 Skills Popularity	33.4	45.0
10 Skills Collab	45.7	55.4
10 Skills Content	64.2	77.5
10 Skills Hybrid	73.6	73.9

TABLE III
Recommending Skills to Individuals - F1-Scores

The next step was using the models to manually rate 5 recommended skills to 10 users. In this the high scores of the Content became obvious. The model recommended skills to users that they already possessed. As a result the metric scores are biased in the case of the Content model and only slightly in the Collaborative model. The reason for this could lay in the fact that the data set is messy and synonyms, non-stemmed words and similar structures such as *Oracle ** are very apparent.

To make the ratings as fair as possible the already acquired skills, non-stemmed words and similar words were ignored. The average scores of the models are as such:

Comparing the manual ratings and the metric scores it becomes obvious that the difference between the Content and Collaborative model are much smaller. The difference lays in the fact that the Content model recommends similar skills and/or skills that share the same word structure. Examples of these are *Civil Aviation*, *Civil Law* and *Civil Litigation*. A lot of emphasis will be put on the word *Civil* which in turn results in the previously stated words. While the Collaborative filtering model recommends more niche skills. Using the same example, the recommended skills were *Public Speaking*, *Teaching* and *Event Management* for the *Civil Aviation* examples. These skills can be considered relevant for the user. Nevertheless, they can be considered to have no association with the term *Civil Aviation*.

The Hybrid model outperforms both due to making recommendations that either both models miss or copying recommendations from the stronger model in a specific situation. As a result the marginally higher rating score is achieved.

Model	Average Score
Popularity	2.24
Content	3.52
Collab	3.28
Hybrid	3.62

TABLE IV
Average Rating Scores of each Model based on At Least 3 Skills

B. Recommend people based on skills

The following table shows the metric scores of the approach recommending individuals based on key skills. For this the data set was limited to the amount of times it shows up within the data set. In this case the skill needed to interact with at least 3 or 10 persons.

Model	Precision @5	Precision @10
3 Persons Popularity	20.3	30.2
3 Persons Collab	23.9	35.7
3 Persons Content	54.5	65.8
3 Persons Hybrid	92.3	99.2
10 Persons Popularity	21.2	30.3
10 Persons Collab	25.1	36.3
10 Persons Content	47.2	60.3
10 Persons Hybrid	91.6	99.0

TABLE V
RECOMMENDING INDIVIDUALS BASED ON SKILLS - PRECISION SCORES

Popularity and Collaborative are similar in scores, just like in the previous section. Hybrid outperforms all other models by a large difference. There is no difference between the amount of times a skill interacted with an unique person. As a result both data set constructions can be used to make recommendations in the case of the Hybrid model. However, when compared to the recommend skill models these scores are much lower with the non-ensemble models. A reason for this could be the fact that Hybrid was able to find candidates that both the Collaborative and Content model missed.

Model	Recall @5	Recall @10
3 Persons Popularity	9.1	13.4
3 Persons Collab	13.4	20.0
3 Persons Content	53.0	64.1
3 Persons Hybrid	55.0	59.0
10 Persons Popularity	9.6	13.7
10 Persons Collab	14.8	21.5
10 Persons Content	45.8	58.5
10 Persons Hybrid	56.9	61.6

TABLE VI
RECOMMENDING INDIVIDUALS BASED ON SKILLS - RECALL SCORES

The recall scores are very low in the case of the Popularity and Collaborative models when compared to the Content and Hybrid models and the precision scores. While the Content scores can be considered good when compared to the other models and its precision scores. In this situation there is a clear difference between the 3 and 10 person data set, a difference of 5.6 percent in the case of 10 hits. However, a reason for this is, is difficult to find when compared to the recall problem. Because the relevancy can drop when the recommended users are being analyzed with more features such as the skill list, location, job position and educational background. Purely based on interpretation the recall scores can vary greatly.

The F1-Scores are within expectations based on the precision and recall tables. As can be seen the scores are much lower when compared to the recommended skills model. This can be due to the fact that finding persons for specific skills is much more complicated. Because more variables need to be taken into account when recommending people, examples of these are location, educational background and experiences. While, recommending skills only looks at the current skill list of the user. With this the recommending model can be suited for finding specific people.

Model	F1-Score @5	F1-Score @10
3 Persons Popularity	12.5	18.6
3 Persons Collab	17.2	25.6
3 Persons Content	53.7	64.9
3 Persons Hybrid	68.9	74.0
10 Persons Popularity	13.2	18.8
10 Persons Collab	18.7	27.0
10 Persons Content	46.5	59.4
10 Persons Hybrid	70.2	75.9

TABLE VII
RECOMMENDING INDIVIDUALS BASED ON SKILLS - F1-SCORES

The biggest problem in this recommendation approach is the vectorized string in the Content Based model. In the recommender skills to people approach, skills that are very similar in word structure are spotted very easily and the similarity can even be manually spotted. When it comes to recommending people based on Skills the bottleneck lays in the messy data and the available strings for vectorization. All the user features could be used, however due to varying string length and empty spaces only the first-name column value was used. Due to the fact that it still needed to be an unique string structure that could be traced back and combining other features with this users resulted in less columns in the vectorization due to names with similar names such as *Mohammed*, *Mohammad* and *Mohamed*. Another problem that showed up was the empty values in the case of the skills column in which some people had only 3 skills while others had around 15. As a result the model was unable to handle the varying length. In the next section the recommended persons will be manually rated based on relevancy and network position.

C. Graph Distance

In this manual rating section a case study approach was used to make sure the graph distances and similarities between the recommended persons can be traced back. The second reason was to check if people that are completely similar to the recruiter are spotted more easily while people who are completely different are less likely to be known to the user.

In this situation the recruiter is an Sr. Software Engineer focused on Business Technology. As a result the key skills that will be focused on are technological and contrasting skills like lifestyles and policy analysis. This results in the following average recommendation scores and the likelihood of knowing a specific person.

Model	Average Score
Popularity	2.55
Content	2.7
Collab	3.15
Hybrid	3.15

TABLE VIII
Average Rating Scores of each Model based on Technological Skills - At least 3 Persons with that Skill

The skills used in the technological section were: Graphic Design, Unix, C and Business Intelligence. In this situation Popularity actually shows expected results due to the fact that the most popular persons will always be chosen. Based on bias, focus of the data set or specific weights this could lower or increase the score. The Content average recommendation score is as expected due to the evaluation not being as accurate as the skills section, due to the strings not being as well fitted as the recommended skills approach. More preprocessing is necessary to make the Content model more accurate.

The Collaborative and Hybrid share the same score, which is interesting due to the fact that Hybrid combines both Content and Collaborative recommendations. This means that the Hybrid approach does not become lower due to the influence of the Content model. The Collaborative score is as expected due to the user-item matrix just being a transposed version of the recommend skills approach.

The next step was checking the path variation between the possible candidates and the recruiter. Where 5 is a direct path, 3 is an indirect path through shared relationships and 1 is no path at all. The method for this was the traditional approach when it comes to recruitment. Where you try to find the person and check how you can contact them, either directly, through a relationship or cold-calling.

Model	1	3	5
Popularity	16	4	0
Content	16	3	1
Collab	17	3	0
Hybrid	17	2	1

TABLE IX
Degree variation between the models Average Rating Scores of each Model based on Technological Skills

The difference between the model results are very close to each other. This could be attributed to the high threshold of 0.3 and the messy data set. Lowering the threshold could increase the amount of indirect and direct paths. However, this cannot be done with the current data set due to the fact that more preprocessing and more unique values are necessary.

The next skills are the contrasting key skills such as Lifestyle, Fundraiser and Policy Analysis. This results in the following scores:

Model	Average Score
Popularity	2.87
Content	3.07
Collab	2.7
Hybrid	3.17

TABLE X
Average Rating Scores of each Model based on Contrasting Skills - At least 3 Persons with that Skill

These results vary a lot with the technological skills. This could be due to the distribution of the data set that focuses more on non-technological skills. This is shown with the Popularity model, which has got a higher score than the technological skills recommendation.

In these results Content and Collaborative are swapped with their technological skills results. This could be again due to the data set having a skewed distribution or the specific key skills selection being more suited for specific models and the dataset. Hybrid is similar to the technological skills results and even is able to use the strengths of Content and Collaborative models to get a higher score than both.

Model	1	3	5
Popularity	24	6	0
Content	30	0	0
Collab	25	5	0
Hybrid	29	1	0

TABLE XI
Degree variation between the models Average Rating Scores of each Model based on Non-Technological Skills

These results are as expected due to the recruiter focusing more on the Technological skills. The shared paths with friends connecting with the possible candidates are expected results, resulting in some indirect paths.

In *Appendix III: Graph Distance of 100 Persons* it can be seen that in the case of 0.1 threshold there is a clear cluster in the middle of people that have direct paths. While 0.3 and 0.5 have either a cluster in the middle of persons possibly knowing each other or just consist of a ring. This shows that performing a social network analysis based on similarity between user features needs to be optimized based on the data set. If the data set consists of many different features a low similarity threshold can be used, while using a dataset, like the one used in this project a higher similarity threshold is needed to prevent similarities based on just one feature.

V. Conclusions

In conclusion, social network analysis and recommender systems can be combined to make the recruiting process as complex or simple depending on the requirements of the recruiter. If a recruiter only looks in it's own social network and company's network than a recommendation skill model can be applied to find people that could fit the requirements of a job offer, if current employees are willing to learn a specific skill. While, if a recruiter wants to look outside their social network results in a more complex model in which the indirect paths and the distance between the nodes need to be taken into account to reach the *"perfect"* applicant.

Based on the results of the models, the recommended skills to people approach is the easiest to implement. Due to the fact that you can limit yourself to specific persons and check if the recommended skills are relevant for them. While the people based on skills approach needs to take a lot more into account and cannot be used with the same model. Due to the fact that the social network aspect is not taken into account. This means that either all models need to incorporate weights. With this users can be removed based on requirements such as locations and distance. The former can be achieved by the Popularity model by adding weights to users based on specific values, this method was not used in the project to prevent one model outperforming the others due to parameter optimization.

The last aspect consists of the term relevancy. Due to the fact that relevancy can vary based on interpretation and that a relevancy score of a specific skill can be given a 1 or 5 based on different persons the results in this project can vary. Other methods to rate relevancy such as binary or multilevel of 3 could have been used in this project. However, the relevancy rating of 1-5 was used to accentuate the differences in rating scores between the models.

VI. Future Work

In the future a collaboration with an application as LinkedIn is necessary to make the model more accurate and to implement the model into different business environments. The data set needs to at least consist of the same values as were used in this project. Additional values are connections with other persons, complete names, entire jobs and educational background, ratings of skill and certificates. If this information is available the mastery of the same skills can vary between people. As a result the recruiter can set a minimum level of mastery of specific skills and can check that based on the rating of the skill and the available certificates.

The models will make the recommendations inside and outside a person's social network more broad or limited. An example of this would be the maximum or minimum distance between two persons. For recruiters this could be interesting because they may not want to look at possible candidates within their own direct paths but want to look at persons outside their own community. However, the possibility of this is dependent on the available data set in the future.

References

[1] Velzen, M. v. (2016, October 27). Social Media belangrijkste middel personeelswerving Employer Branding. Accessed on June 23, 2018, from Emerce: https://www.emerce.nl/research/social-media-belangrijkste-middel-personeelswerving-employer-branding

[2] Gulden, M. (n.d.). Wat is LinkedIn en wat kun je ermee. Accessed on June 23, 2018, from The Marketing Factory: https://themarketingfactory.nl/social-media-kennisbank/linkedin/wat-is-linkedin-en-wat-kun-je-ermee/

[3] B. Wellman and S.D. Berkowitz (1998) *Structural Analysis in the Social Sciences 2: Social Structures: a Network Approach*, Cambridge University Press, Cambridge

[4] S. Mishra, R. Borboruah and S. Rakshit (2014) Modeling of Social Network using Graph Theoretical Approach. *International Conference onf Microeletronics, Circuits and Systems (Micro 2014)*

[5] D.Knoke and J.H. Kuklinski (1982) Network Analysis.*Sage University paper Series on Quantitative Applications in the Social Sciences*. no. 07-028, Sage, Newbury Park, California

[6] C. Wetherell, A. Plakans and B. Wellman (1994), Social networks, kinship and community in Eastern Europe, *Jornal of Interdisciplinary History*, vol 24, pp 645

[7] H.D. White (2000), Toward ego-centered citation-analysis. In B. Cronin and H.B. Atkins (eds), *The Web of Knowledge*. Information Today, Medford, New Jersey, pp. 475-496

[8] L.A. Cuttilo, R. Molva and T. Struffe (2009) Privacy preserving social networking through decentralization. *IEEE WONS*

[9] Shani G., Gunawardana A. (2011) Evaluating Recommendation Systems, In Ricci F., Rockach L., Shapira B. , Kantor. P (eds) *Recommender Systems Handbook*. Springer, Boston, MA

[10] Isinkaye F., Folajimi Y. and Ojokoh, B. (2015) Recommendation systems: Principles methods and evaluation, *Egyptian Informatics Journal*, pp. 261-273

[11] Herlocker J., Konstan J.A. , Terveen L.G. and Riedl T. (2004) Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* pp 5-53

[12] Burke R. (2002) Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, Vol 12, Issue 4. pp 331-370

[13] Adomavicius G. and Tuzhilin A. (2005) Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions: Knowledge and Data Engineering* Vol 17, Issue 6. pp 734-749

[14] Ekstrand M.D. , Riedl J.T. and Konstan J.A. (2011) Collaborative Filtering Recommender Systems. *Foundations and Trends in Human-Computer Interaction* Vol 4, Issue 2. pp 81-173

[15] Schafer J.B., Frankowski D., Herlocker J. and Sen S. (2007) Collaborativer Filtering Recommender Systems. In: Brusilovsky P., Kobsa A. Nedjl W. (eds) *The Adaptive Web. Lecture Notes in Computer Science* Vol. 4: No. 2, pp 81-173.

Appendix I: Mathematical Formulas

The following mathematical algorithms were used during the execution of this project.

Precision

$$\frac{RelevantDocuments \cap RetrievedDocuments}{RetrievedDocuments}$$

Recall

$$\frac{RelevantDocuments \cup RetrievedDocuments}{RelevantDocuments}$$

F1-Score

$$2 * \frac{precision * recall}{precision + recall}$$

TF-IDF

$$W_{ij} = tf_{ij} * log(\frac{N}{df^1})$$

Linear Kernel

$$k(x, y) = x^T y$$

Appendix II: Graphs

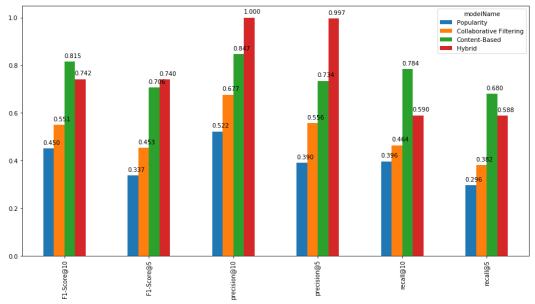

Fig. 6. Recommend Skills to People more than 3 acquired Skills

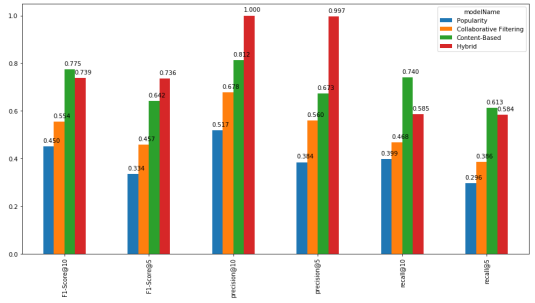

Fig. 7. Recommend Skills to People more than 10 acquired Skills

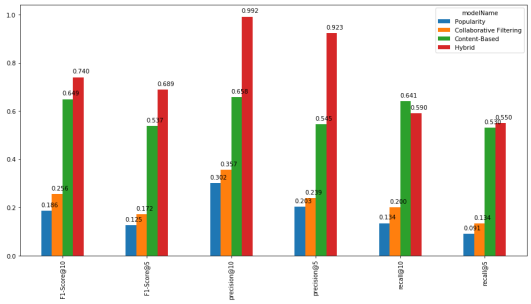

Fig. 8. Recommend People based on Skills who appear more than 3 times

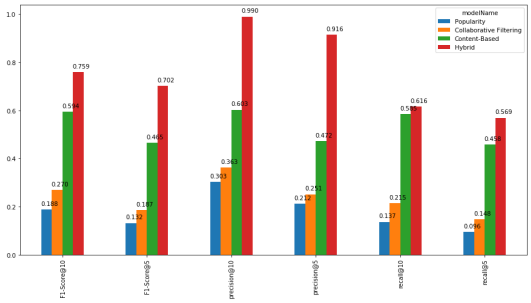

Fig. 9. Recommend People based on Skills who appear more than 10 times

Appendix III: Graph Distance of 100 Persons

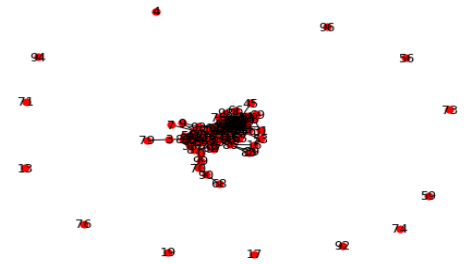

Fig. 10. Similarity Matrix of the first 100 Persons in the data set - Threshold of 0.1

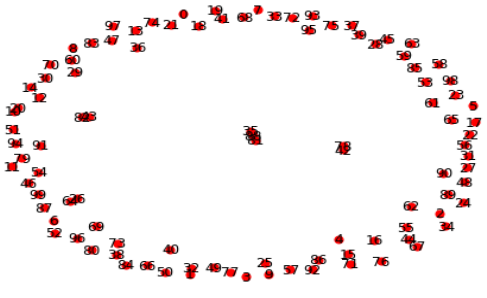

Fig. 11. Similarity Matrix of the first 100 Persons in the data set - Threshold of 0.3

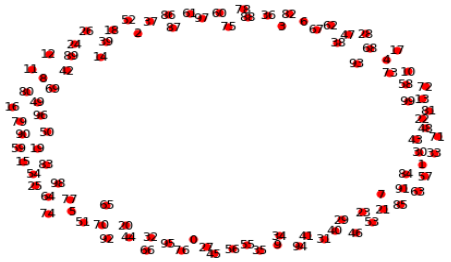

Fig. 12. Similarity Matrix of the first 100 Persons in the data set - Threshold of 0.5

Appendix IV: Author's Contribution

The author's contribution in this project can be split into individual and shared contributions.

The shared contributions were collecting and preprocessing the LinkedIn scrape. Brian preprocessed and converted the JSON to CSV. Jordi preprocessed the values and removed the nan values, incomplete columns. The manual rating of the recommended skills and persons were done by both person. As to prevent personal bias to the relevancy score.

Brian's contribution consisted of coding the similarity matrix.

Jordi's contribution consisted of changing a pre-existing recommender system, calculating the distance between the case study recruiter and possible candidates.

The report sections were written in this way:

- Abstract: Jordi Beernink
- Introduction: Brian Westerweel
- Background SNA: Jordi Beernink
- Background Graph Theory: Jordi Beernink
- Background Recommender Systems: Jordi Beernink
- Background Implementation: Jordi Beernink
- Experiment Approach: Jordi Beernink
- Experiment Dataset: Jordi Beernink
- Experiment Evaluation: Jordi Beernink
- Results Recommending Skills to People: Jordi Beernink
- Results Manual Rating Skills to People: Shared
- Results Recommending People based on Skills: Jordi Beernink
- Results Manual Rating People based on Skills: Shared
- Results Graph Distance: Jordi Beernink
- Conclusion: Jordi Beernink
- Future Work: Jordi Beernink

Appendix V: Github Repository

https://github.com/deprehend0/CCML2018/tree/master/Project