

A scalable recommendation system approach for a companies - seniors matching

1st Kévin Cédric Guyard
Information Science Institute, GSEM/CUI
University of Geneva
Geneva, Switzerland
kevin.guyard@unige.ch

2nd Michel Deriaz
HEG Genève
HES-SO
Geneva, Switzerland
michel.deriaz@hesge.ch

Abstract—Recommendation systems are becoming more and more present in our daily lives, whether it is for suggesting items to buy, movies to watch or music to listen. They can be used in a large number of contexts. In this paper, we propose the use of a recommendation system in the context of a recruitment platform. The use of the recommendation system allows to obtain precise profile recommendations based on the competences of a candidate to meet the stated requirements and to avoid companies to have to perform a very time-consuming manual sorting of the candidates. Thus, this paper presents the context in which we propose this recommendation system, the data preprocessing, the general approach based on a hybrid Content-Based Filtering (CBF) and Similarity Index (SI) system, as well as the means implemented to reduce the computational cost of such a system with the increasing evolution of the platform.

Index Terms—Artificial intelligence, machine learning, recruitment, hybrid recommendation system, collaborative filtering, similarity index.

I. INTRODUCTION

This paper is proposing a recommendation system in the frame of the platform “WisdomOfAge”. It is a web platform currently under creation which is aiming to connect senior workers who will be retired or are already retired with companies. On one hand, seniors fill their resume to describe their skills, and on another hand, companies can create mission offers. This platform has dual goals. For seniors, this will allow them to stay active and keep a social link. For companies, this will allow them to acquire internally missing skills without hiring or contracting consulting companies which are often expensive. The recommendation system will be used to generate a list of seniors according to the company’s needs when a company is creating a new mission offer on the website.

This frame is leading to several specific problems for the recommendation system.

The recommendation system should be operational and able to propose recommendations starting from the first mission offer on the website (with the condition that there is at least one senior registered on the platform corresponding to the need of the company). The system should be scalable and able to self-adjust to the number of users registered on the website.

This work was co-funded by the State Secretariat for Education, Research and Innovation of the Swiss federal government and the European Union, in the frame of the EU AAL project WisdomOfAge (AAL-2020-7-83)

Another difficulty is the fact that seniors have often a career which has evolved during their professional life. An experience done in the beginning of the career is less relevant than an experience done at the end of the career. The recommendation system should be able to detect if experience / skill is relevant to provide suitable recommendations to companies.

The platform will be available in different languages. Companies and seniors will be matched only if they share the same language but the recommendation system must be able to deal with text from different languages.

II. STATE OF THE ART

In their paper [1], Anushka Agarwal and Dr. Senthilkumar present a resume-job recommendation system based on similarity index. They have used cosine similarity to measure the similarity between a job offer and a resume to rank resume using the similarity score.

François Mentec, Zoltán Miklós, Sébastien Hervieu and Thierry Roger propose in their publication [2] a conversational recommendation system to recommend resumes regarding a job offer. Their approach seems interesting since their system is able to discuss with the recruiter to refine the recommendation according to the requirements of the recruiter. However, their system needs a dataset to be tuned whereas WisdomOfAge does not have any data for now. Moreover, the platform does not give the possibility to discuss with companies to refine recommendation but it could be an interesting system if this feature could be developed.

In [3], Ravita Mishra and Sheetal Rathi propose a survey of recommendation system models used in important recruitment platform like LinkedIn.

In [4], Shaha T. Al-Otaibi and Mourad Ykhlef present a technical survey in the field of job recommendation system.

In [5], Pradeep Kumar Roy, Sarabjeet Singh Chowdhary, Rocky Bhatia propose an approach based on a classifier preceding a hybrid system using a content-based filtering (with cosine similarity) and a similarity index (using kNN).

In [6], Demirtsoğlu Georgios, Zisopoulos Xarilaos, Karagiannidis Savvas present content-based filtering systems and then discuss of their advantages and their drawback. They also present concrete examples currently in place.

In [7] F. Amato et al. propose a comparison of rules based, support vector machine classifier and Latent Dirichlet Allocation (LDA) to provides a classification of job offers.

As we can observe in the literature, authors suggest several ways to propose recommendations in the field of the recruitment.

In one hand, some authors suggest the use of multiclass classifier model to achieve a classification of the candidates' profile. Thus, recruiters can filter candidates on the category attributed by the model. However, it does not allow them to research for a particular profile. Indeed, all candidates who belong the same category are considered as "equal", there is no notion of ranking. Then, the recruiter should make his own ranking on the first filtered set.

In other hand, some of them propose systems which provide a ranking of the candidates based on the similarity between their profiles and the job offer.

In this paper, we suggest a recommendation system based on the similarity of the mentors' profile and the mission offers. Moreover, the system also use feedback let by precedent companies to increase its recommendation accuracy.

III. DATA

To provide recommendations, the system has access to:

- Companies' profiles: activity sector, description
- Mentors' profiles: educations (dates, degree, faculty, description), experiences (dates, position, description), skills
- Mission offers: title, category, description, keyword(s), discipline(s)
- Feedback: rating (0 to 10), description

IV. PREPROCESSING

In first place, it is important to note that it is crucial to preprocess the text. Indeed, this step is important to avoid a decrease of performances in most of cases for every machine learning system. In his paper [8], Ammar Ismael Kadhim have proposed the following steps: tokenization, stop words removal, stemming, text document representation and feature extraction (see Fig.1). This preprocessing pipeline can be found in numerous text applications and has been already validated.

Tokenization is a way of separating a piece of text into smaller units called tokens. Here, tokens can be either words, characters, or sub words. Hence, tokenization can be broadly classified into 3 types – word, character, and sub word (n-gram characters) tokenization [9].

The goal of removing stop words is to minimize the impact of common and worthless words . The list of stop words is depending on the language (e.g., in English, stop words would be "the", "a"...). Although in some cases, it could be

interesting to remove rare words, they will not be removed in our context since rare words could outline rare skills.

Stemming step is a replacement of each token by its stem (it is the token in which prefix and/or suffix is removed). Nevertheless, there are some discussions to use lemmatization instead of stemming. Lemmatization is taking the morphological form of the word into account, based on a dictionary. The authors Divya Khyani, Siddhartha B S, Niveditha N M, Divya B M suggest in their paper [10] that lemmatization tends to better keep the meaning of the words but is more complex to be implemented.

It is also important to note that, depending on the language, stemming and lemmatization can produce interpretation errors inherent to the principle on which is based each concept. Thus, [11] provides examples of these errors. We can note that in Telugu language, stemming generate the same stem for words "robe" and "I don't share". In Greek language, lemmatization can generate the same lemma for words "imperfective" and "perfective".

It would be interesting to study each language case to determine for which languages lemmatization gives the better performance and for which stemming performs better.

With text document representation, a numerical vector from a text is generated. A key element of working with common machine learning algorithm is that vectors should have the same length regardless of the text size of the document. Our system is using the text document representation called bag of words (BoW). For each document, a vector of the size of the vocabulary present on our whole corpus (here all mission offers, companies' profiles and seniors' profiles) is generated. Each box represents the number of occurrences of a token in the document.

There are also other text document representations such as GloVe [12], FastText [13], BERT [14] etc. However, these representations are not available in all languages while WisdomOfAge is aiming to expand to several European languages. It will still be appropriate to study these possibilities to check if their implementation is or is not a possible option and if our system would benefit of using them.

The last step called feature extraction aims to show up the important features. Indeed, BoW only counts the number of occurrences of each token in a document. However, if a company or a senior search for / own a particular skill, the mission offer / senior profile will contain particular words which are not used frequently in the corpus. Thus, the idea is to weight the BoW using higher weight for rare words in the corpus. The weight will be computed using the inverse document frequency. It will allow our system to detect rare or particular skills and extract them from the mass of the pool of mission offers / senior profiles.

Note that text document representation using BoW in addition with feature extraction using inverse document frequency is the famous called term frequency – inverse document frequency (TF-IDF).

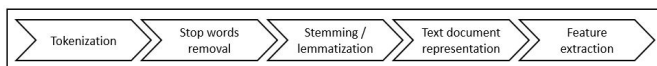


Fig. 1. Preprocessing pipeline

V. GENERAL APPROACH

The recommendation system proposed in this paper is a hybrid recommendation system. It is constituted of recommendation sub systems whose outputs are assembled. Indeed, each recommendation system has strengths and drawbacks. Using several sub systems decreases the drawbacks of the general system to get more elaborate recommendations. There are several techniques to assemble the results of each sub system. This point will be discussed in section 5.D. Hybrid recommendation system.

A. Collaborative Filtering

CF recommendation systems are very popular in the field of recommendation system. Indeed, they allow accurate recommendations. However, they suffer of the cold-start problem. This point will be discussed in section 5.D. Hybrid recommendation system. CF can be divided in two categories: User-based Collaborative Filtering (UCF) and Item-based Collaborative Filtering (ICF).

The first category, UCF, is based on the idea that if two users are similar, they have similar preferences. Thus, systems based on this principle try to find similarity between users. This similarity can be established using one or several metrics on the features described by users. The system then provides a recommendation list from the items liked by similar users.

The second category, ICF, base its recommendation concept on the fact that if a user likes an item, he is willing to like similar items. ICF systems try to find similarities between items (through features describing items) to propose the more similar items of the items liked by the user.

In our context, users are companies posting mission offers. The features which describe these users are the elements of the companies profiles and the mission offer. On the other side, items are the seniors registered on the platform, with their features extracted from their profiles.

From the concept of ICF, we can conclude it is not adequate to reach our goals, for the following reasons: the first one is that if a company enjoyed working with a senior and has a similar need later on, there is a very high probability that the company will not use the platform to ask for a recommendation list of seniors. Indeed, the company already owns the list of recommendation for its first mission with contact details and will not pay a second time for the same information. If the company has a different need, there is a low probability that the senior previously selected for the first mission will fulfil requirements for the second mission. Moreover, senior profiles include only technical features. Thus, it is not possible for now that our system understands the personality part of the senior that the company has enjoyed. So, without addition of these nontechnical features, ICF concept is not interesting in our context.

On the other hand, UCF systems seems to be more suitable to our needs. Indeed, if a company has enjoyed working with a senior, there is a high probability that this senior is a good choice for a similar company with a similar need. In this way, the choice to focus on the concept of UCF is fully logical. To

do this, we compute the similarity metrics to find the more similar user to the company which request a recommendation for a mission offer and weight the similarity regarding the feedback level given by the other user at the end of their missions with this user.

B. Similarity Index

The proposed SI recommendation system is comparing senior profiles with the mission offer of the company. It is about to provide a list of seniors with the highest similarity to the mission offer. To achieve this, each element of the senior profile is compared one by one to the mission offer (each skill, each education, each experience).

For the skill part, the similarity is weighted by the level assigned to the skill by the senior. However, this level is not treated in an absolute way but in a relative way. Indeed, evaluation of skill can differ from one person to another. Thus, the weight credited to the skill is relative to the weights that the senior has credited to these other skills. The idea is to detect more important skill of the senior.

For the education and experience part, each similarity is weighted by a weight which is computed depending on the duration of the education/experience and on the time from when the experience ended.

C. Latent Dirichlet Allocation

As stated in the introduction, the recommendation system should be scalable. Moreover, it should be able to work in a classic server without particular calculation power. Up to here, the two proposed subsystems can work properly with a limited number of persons registered on the platform. However, with the expected growth of the platform, these solutions could be too greedy in calculation resources to work within acceptable time frame.

Thus, to keep the time to return the recommendations reasonable, companies, seniors and mission offers will be daily clustered using LDA which is a soft clustering method, that is elements are not part of one cluster but belong to all clusters with a specific percentage.

When a company will post a new mission offer, the percentage of subscription to each cluster is computed. Then, the two subsystems do not work with all the data of the platform but only those from the clusters which are the more representative of the mission offer (in other term, the clusters for which the percentage of subscription is the highest).

D. Hybrid recommendation system

In one hand, CF is more likely to produce accurate recommendation. Indeed, it does not base its recommendations only on objectives and mentors' description but also on the feedback given by companies. This principle allows CF to capture more information from the phenomena. However, CF suffers from cold start. It is not able to give recommendations without feedback at the beginning and is not able to recommend a new mentor who does not have feedback. In other hand, SI is less accurate but will not suffers from the cold start.

Thus, we suggest to combine both CF and SI in a hybrid recommendation system to overcome the drawbacks of both systems. We propose a weighted sum to combine the output of each subsystem. In other words, the output of each subsystem is weighted and summed to obtain a unique score to rank mentors. The overview of the complete system is proposed in Fig.2.

The use of a weighted sum and not a equal weight sum is important to control the weight given to each subsystem along the life cycle of the platform. Indeed, at the beginning, CF will not be able to propose a lot of accurate recommendations since there will be only few feedback. But with the growth of the platform, CF will become able to give complete recommendations list without the help of the SI. In this way, new mentors will never be recommended. Thus, decreasing the weight of the CF over time will overcome this problem.

VI. SIMILARITY MEASURE

The two subsystems measure similarity between two vectors which represent text information. To measure this similarity, the most known solution is the cosine similarity. However, some authors suggest noticeable improvements with the use of more elaborate methods. For example, Shereen Albitar, Sébastien Fournier and Bernard Espinasse suggest in their paper [15] the use of SemIDF or SemTFIDF for measuring the similarity between two vectors which represent text in the context of a classification task. It allows them to achieve better performances than with simple cosine similarity. Thus, it will be necessary to assess these methods to decide if they could lead to an improvement of our system performances.

After similarity evaluation, the score is weighted according to the element. Indeed, it can be relevant to apply different weights for the different elements of comparison. As part of CF, it corresponds to assess the importance of the description of the company, its sector of activity and the description of the mission offer. For the SI, it corresponds to assess the importance of skills, educations and experiences.

VII. SYSTEM EVALUATION

A. Labeled data

One problem is that we are facing of a lack of labeled data to evaluate the model. Indeed, the platform being in development, we do not have past data of its activity. To collect labeled data,

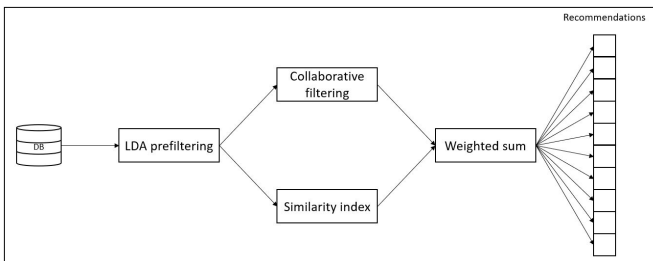


Fig. 2. Recommendation system synoptic

the platform will add a feature to allow companies to label data in exchange of a discount on the website. Concretely, when a company will accept to participate, the user will be asked to create an objective linked to its activity sector. After this step, a long list of mentor's profiles will be exposed and the company will label each mentor regarding its skills to achieve the objective (label could be "relevant" or "irrelevant").

B. Evaluation metrics

It exists a large range of metrics to evaluate the classification results of recommendation systems. These metrics can be classified into two categories: Decision support metrics and ranking-based metrics. In the first one, the ranking of the item is not taken into consideration. In the second, errors are more penalized if there are in the top of the recommendations. Since the platform will propose the mentors' profiles in a vertical list, companies will probably start by taking a look on the first mentor, then the second etc. In this way, using a ranking-based metric is necessary. The Mean Average Precision (MAP) is a good choice regarding our context.

First, we define the Average Precision (AP) to evaluate a recommendations list for a single objective:

$$AP@N(o) = \frac{1}{\min(m(o), N)} \sum_{k=1}^{\min(m(o), N)} P(k) * rel(k) \quad (1)$$

where:

- o is the given objective for which we request recommendations
- N is the number of recommended mentors by the recommendation system
- $m(o)$ is the number of relevant mentors in the dataset for the given objective
- $P(k)$ is the number of relevant mentors in the k first recommended mentors divided by k (known as precision@ k)
- $rel(k)$ is an indicator function. It is equal to 1 if the k^{th} recommended mentor is relevant, 0 otherwise

Note that we have slightly changed the AP formula. Indeed, the business plan of the platform imposes to give N recommendations for each request as long as there are at least N relevant mentors registered on the platform regarding the request. Thus, we have changed the N in the initial formula by $\min(m(o), N)$ to ensure that the metric does not penalize a request for which there is not enough relevant mentors to achieve a complete recommendations list of size N .

We can define the MAP to evaluate the recommendations list for all our test objectives:

$$MAP@N(O) = \frac{1}{size(O)} \sum_{o \in O} AP@N(o) \quad (2)$$

where:

- O are the given objectives for which we request recommendations (one by objective)

In addition to the MAP which evaluate the classification results of our system, we will also introduce another metric: the coverage. The coverage is the total number of recommended mentors for all our test objectives divided by the total number of mentors available in the platform. This metric is important to ensure that our recommendation system will not always recommend the same mentor (even if these mentors are relevant).

Since we will have a small dataset, it will be impossible at the beginning to have a complete coverage since some mentors could be irrelevant for all our test objectives. Thus, we will focus to maximize in first place the MAP while keeping a look on the coverage to ensure that a little improvement on the MAP will not induce an important decrease of the coverage.

VIII. CONCLUSION

In this paper, we have proposed a general approach for a scalable recommendation system in a particular context. This context raises specific issues involving technical choices. To overcome these constraints, we suggest a hybrid recommendation system based on a collaborative filtering and a similarity index. During the implementation of the system, it will remains to bring answers to the pending choices, as for example for the choice between stemming and lemmatization according to the languages or the interest of more sophisticated similarity measure than the cosine similarity.

REFERENCES

- [1] Anushka Agarwal, Dr. Senthilkumar, Resume recommendation system using cosine similarity, International Research Journal of Modernization in Engineering Technology and Science, Volume:04/Issue:04/April-2022
- [2] MENTEC, François, MIKLÓS, Zoltán, HERVIEU, Sébastien, et al. Conversational recommendations for job recruiters. In : Knowledge-aware and Conversational Recommender Systems. 2021.
- [3] MISHRA, Ravita et RATHI, Sheetal. Efficient and scalable job recommender system using collaborative filtering. In : ICDSMLA 2019. Springer, Singapore, 2020. p. 842-856.
- [4] SHAHA, T. Al-Otaibi et MOURAD, Ykhlef. A survey of job recommender systems. International Journal of Physical Sciences, 2012, vol. 7, no 29, p. 5127-5142.
- [5] ROY, Pradeep Kumar, CHOWDHARY, Sarabjeet Singh, et BHATIA, Rocky. A Machine Learning approach for automation of Resume Recommendation system. Procedia Computer Science, 2020, vol. 167, p. 2318-2327.
- [6] ZISOPOULOS, Charilaos, KARAGIANNIDIS, Savvas, DEMIRT-SOGLOU, Georgios, et al. Content-Based Recommendation Systems, 2008.
- [7] F. Amato et al., "Challenge: Processing web texts for classifying job offers," Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015), 2015, pp. 460-463, doi: 10.1109/ICOSC.2015.7050852.
- [8] KADHIM, Ammar Ismael. An evaluation of preprocessing techniques for text classification. International Journal of Computer Science and Information Security (IJCSIS), 2018, vol. 16, no 6, p. 22-32.
- [9] <https://www.analyticsvidhya.com/blog/2020/05/what-is-tokenization-nlp/>
- [10] KHYANI, Divya, SIDDHARTHA, B. S., NIVEDITHA, N. M., et al. An Interpretation of Lemmatization and Stemming in Natural Language Processing. Journal of University of Shanghai for Science and Technology, 2021.
- [11] <https://blog.bitext.com/what-is-the-difference-between-stemming-and-lemmatization/#:~:text=Stemming%20and%20lemmatization%20are%20methods,the%20word%20is%20being%20used.>
- [12] PENNINGTON, Jeffrey, SOCHER, Richard, et MANNING, Christopher D. Glove: Global vectors for word representation. In : Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014. p. 1532-1543.
- [13] BOJANOWSKI, Piotr, GRAVE, Edouard, JOULIN, Armand, et al. Enriching word vectors with subword information. Transactions of the association for computational linguistics, 2017, vol. 5, p. 135-146.
- [14] DEVLIN, Jacob, CHANG, Ming-Wei, LEE, Kenton, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
- [15] ALBITAR, Shereen, FOURNIER, Sébastien, et ESPINASSE, Bernard. An effective TF/IDF-based text-to-text semantic similarity measure for text classification. In : International Conference on Web Information Systems Engineering. Springer, Cham, 2014. p. 105-114.