Neural Weak Supervision Model for Search of Specialists in Scientific Data Repository

Sergio Jose de Sousa[1(✉)], Thiago Magela Rodrigues Dias[1], and Adilson Luiz Pinto[2]

[1] Departamento de Modelagem Matemática e Computacional, Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), Belo Horizonte, MG, Brazil
thiagomagela@cefetmg.br

[2] Departamento de Ciência da Informação, Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brazil
adilson.pinto@ufsc.br

Abstract. With the growing volume of data produced today, it is clear that more and more users are using different types of systems, such as, for example, professional and academic data storage systems. Given the large amount of stored data, the difficulty of finding candidates with appropriate profiles for a particular activity is noteworthy. In this context, to try to solve this problem comes the expertise retrieval, a branch of information retrieval, which consists of, given a query, documents are recovered and used as indirect units of information for the candidates and some aggregation techniques are used in these documents to generate a score to the candidate. There are several models and techniques to work with this problem, some have been tested extensively but the search for specialists in the academic field with neural models has a smaller amount of research, this fact is due to the complexity of these models and the need for large volumes of data with judgments of relevance or labeled for your training. Therefore, this work proposes a technique of expansion and generation of weak supervised data where the relevance judgments are created with heuristic techniques, making it possible to use models that require large volumes of data. In addition, is proposed a technique of deep auto-encoder to select negative documents and finally a ranking model based on recurrent neural networks and that was able to overcome all the baselines compared.

Keywords: Expertise retrieval · Deep learning · Weak supervision.

E. Bisset Álvarez (Ed.): DIONE 2021, LNICST 378, pp. 286–296, 2021.
https://doi.org/10.1007/978-3-030-77417-2_21

1 Introduction

The challenges of finding the required information increases with the volume of data. Therefore, support tools such as recommendation and information retrieval systems are essential when searching, whether searching for websites, shopping website recommendations, searching for specialists, among other possibilities.

Search for specialists has existed since before the invention of the computer and denotes the need to find someone with some specific knowledge. In the field of psychology, in [4] it is said that the superior performance of specialists is acquired through experience, repetitions and structuring of long-term activities. For [15] specialists are people with knowledge or who have mastered detailed skills in specific areas. This task is challenging because it is necessary to evaluate what the person has already done, worked and produced in order to find out what their specialties are and who stands out among various options.

The sub-domain expertise retrieval in academic environment is one of the areas that has been receiving more and more attention despite having less related research [2], which may be related to the complexity of the problem or the difficulty of obtaining relevant data on the topic. In the literature it is possible to find some bases such as in [1,3,23] where it is possible to observe expert datasets with a focus on only one topic, not being generalist, centralized in just one organization or in only one area. Some platforms for sharing and storing projects and academic works that stand out, for example Academia[1], DBLP[2], Google Scholar[3], Plataforma Lattes[4], Microsoft Academic Search[5], ResearchGate[6].

Specifically, the *Plataforma Lattes* developed and maintained by the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq) appears as an important source of academic knowledge. In [6] the authors demonstrated how relevant the data contained in the platform are for the understanding Brazilian research and science, including personal, professional and academic information, such as scientific and technological production. Therefore, the *Plataforma Lattes* is an expressive source of high quality information from individuals [13] and is a good source of data to feed models and techniques in general.

For any Information Retrieval (IR) system, the relevance of the items returned given a query is extremely important. These measures may vary according to the robustness, sensitivity and efficiency that the system is expected to demonstrate [18]. These attributes help us to compare and select traditional and neural models that recently gained prominence.

The first artificial neural network proposed in [21] with a very simple format called perceptron that today is basically a neuron unit of current network, it works like a linear binary classifier that tries to find a better separation of

[1] Academia: https://www.academia.edu/.
[2] DBLP: https://dblp.uni-trier.de/.
[3] Google Scholar: https://scholar.google.com/.
[4] Plataforma Lattes: http://lattes.cnpq.br/.
[5] Microsoft Academic Search: https://academic.microsoft.com/.
[6] ResearchGate: https://www.researchgate.net/.

the data. One of the biggest problems of this model is the inability to separate nonlinear data as a xor distribution, to be solved we must increase the depth and width of the network, adding neurons and more layers. This creates other problems, overfitting, when the model decorates training data and fails to generalize validation, another major problem is the need for an advanced hardware architecture that was expensive at the time. But with the reduction of hardware prices like GPUs, the advancement in techniques that reduce overfitting like dropout and maturity and specialties of the architectures provided a great advance and highlight in deep neural network.

Neural networks have brought great improvements in the areas that uses unstructured data like of computer vision, natural language processing and speech recognition [14], unlike traditional techniques, these networks benefit from large amounts of data, having an ability to learn contexts and relationships that are difficult to identify with handcraft methods. More recently, some attempts have been made to propose and adapt these techniques for information retrieval.

In these neural models applied to IR, a major problem stands out, the need for large amounts of labeled data. These labels consist of judgment of relevance, that is, a set of triples containing a query, a document and the score of this relationship. Data labeling is expensive and can take a long time, which makes it impossible to apply these models in many cases.

Motivated by the new neural models and the need for data with judgments of relevance, the present work presents an alternative model with weak supervision where the labels are generated by heuristics, trying to answer the following questions:

Q1: Can weak supervision obtain better results than traditional technique used to generate the judgments of relevance applying to problems of search for specialists?

Q2: Can the generation of negative documents for queries through a deep auto-encoder surpass the standard of selecting random documents?

Q3: Can the Dual Embedding LSTM model surpass the models in the literature?

Therefore, in this work, a technique is proposed to generate pseudo-judgments of relevance to the documents, considering that for a given query it is also necessary samples of relevant documents and samples of non-relevant documents. The response of *Q1* is positive when they combine the technique of weak supervision and the deep autoencoder to select the negative documents. The answer from *Q2* is also positive, the deep autoencoder provides a selection of documents that negatively represent a query more effectively than random selection.

To select samples of positive documents for the queries, a strategy is proposed to generate relevance judgments using classical information retrieval techniques based on language model with Bayesian smoothing using Dirichlet [25] distribution; for samples of negative documents is proposed a deep autoencoder [9], calculating the most distant candidates from each consultation and extracting their documents. To carry out the reclassification of documents, a dual neu-

ral architecture with recurring layers is proposed, in order to recalculate the sequence and scores of the documents that ultimately compose the score of each candidate, answering positively the question *Q3*.

In the next section, it will be presented the Related Works (Sect. 2) to this research, following the applied Methodology (Sect. 3), presenting the entire proposed framework, right after there is a detailed description about the used dataset in Sect. 4 and finally presented the Final Results (Sect. 5) and Conclusions (Sect. 6).

2 Related Works

In [2] an extensive literature review is described that highlights the advances in models and algorithms for searching and ranking specialists, summarizing and establishing the relationships of these approaches. More recently, [11], a survey was conducted that selected 96 articles consisting of 57 journals, 34 conferences and three book chapters, analyzing the domain of expert search, knowledge sources, methods and databases. There was a growing trend in the amount of IR research for searches models by academy experts.

The work [16] use the database LExR the authors proposed a technique based on information theory where the document-author association is given by a probability, that is, a non-Boolean model, and two alternatives normalization schemes which measures how discriminative a particular document-author association is in view of the other associations involved in each author's document. The approach surpassed the proposed baselines.

In [5] the authors proposed a weak supervised model of deep neural network. The tests were carried out on two data-sets, one for news and other with general data from the internet. For the queries, query logs from the service provider AOL[7] were used. Pseudo-judgments of relevance are generated using BM25 [20] with 1000 documents with the highest score being selected for each query and 1000 other negative documents sampled at random. In the experiments three network architectures are proposed, one point-wise and two pair-wise, three representations of the input data, being a dense representation with several calculated attributes, a sparse representation with bag-of-words and finally an embedding vector learned during training. The combination of the pair-wise architecture with representation of embedding as an input surpassed the baseline technique.

The tutorial proposed by [18] presents basic concepts and intuitions behind the neural models applied in IR, reviewing recent neural network architectures, pointing out their positive and negative sides and finally discussing possible future directions.

Part of the hyper-parameters and architecture of the Dual Embedding LSTM model used in this work were inspired by [19] which proposes a duet architecture where the network has a part that learns local representation and second part that learns representation distributed from the sets of queries and documents.

[7] AOL: https://www.aol.com/.

3 Methodology

In this section, the framework for generating the necessary data is presented, such as an extensive list of queries, a list of documents with positive and negative relationships given a query (Sect. 3.1). Next the Dual Embedding LSTM model is described (Sect. 3.2).

3.1 Weak Supervision

Weak supervision is usually a term given to refer to models that have noisy labels but it also refers to models that use labels generated by heuristic techniques [10]. As neural models are greedy with labeled data, it is necessary to obtain pseudo-judgments of relevance, and for correct learning it is necessary for each query a set of documents with a positive relationship and a set of negative documents, that is, results that would be incorrect for a given query. The following are techniques for extracting, given a query, obtaining positive and negative documents together with their labels or relevance judgments. That is,

$$S = f(q, d),$$

where S means the pseudo-judgment score, q the query and d document, with documents varying between relevant $+d$ and no relevant $-d$.

Positive Documents. To select positive documents together with a pseudo-label, the technique based on Language Model with smoothing and distribution of Dirichlet [25] was used as a calculation of similarity between documents and queries. Thus, for each query, up to 20 documents with the best scores are selected.

The Bayesian smoothing with Dirichlet distribution can be seen in Eq. 1, where it tries to estimate the smoothed probability of finding the term i given the probabilities model of the document j ($P^s(i|\theta_{d_j})$) where $tf_{i,j}$ is the frequency of term in the document, F_i the frequency of term i in all documents of corpus divided by the sum of the frequencies of all terms in all documents. Sum up these values and divide by the sum of the frequencies of all the terms of the document j and these terms are weighted by the constant λ ranging from 0 to 1, the closer to 1 the more smoothed the language model becomes.

$$P^s(i|\theta_{d_j}) = \frac{tf_{i,j} + \lambda \frac{F_i}{\sum_i F_i}}{\sum_i tf_{i,j} + \lambda} \tag{1}$$

Negative Documents. To find the most distant documents of each query a deep auto-encoder is proposed in this work. To learn a representation of the candidates in a reduced and latent dimension space [22], the queries are also transformed in this new dimensional space and with that, the cosine similarity between the query and each candidate is calculated, thus selection the most

distant candidates and we extract their documents as negative samples equaling the number of positive and negative documents for each query.

In Fig. 1 it is possible to see the used architecture, having a total of 6 layers, the first 3 being the encoder. The idea is to train the model to reconstruct the data in order to obtain a compact and latent representation. To summary, this process goes through the steps:

1. Documents are grouped by author, generating a bag-of-words vector for each candidate.
2. This vector is used in the training of the auto-encoder.
3. Reconstruction error calculated by the cosine similarity between the input vector and the vector returned by the auto-encoder output.
4. Once trained, the encoder is used to transform all candidates and queries into a reduced dimensional space.
5. Cosine similarity is applied between each query with each candidate.
6. From the most distant candidates, their documents are extracted.

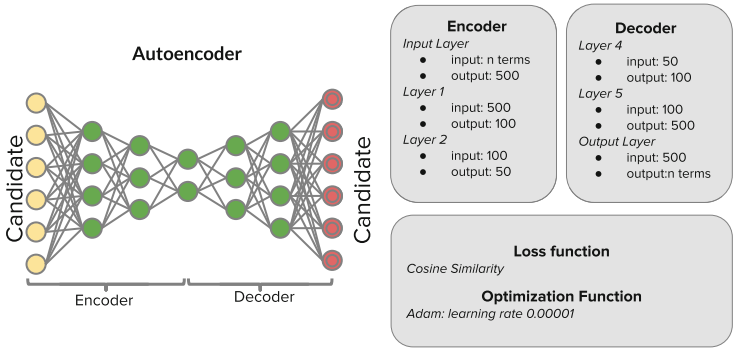

Fig. 1. Deep autoencoder architecture. The first half being called the encoder and the second decoder, next to it you can see the configuration of the proposed neural network.

3.2 Dual Embedding LSTM

As can be seen in Fig. 2, the Dual Embedding LSTM architecture has two input nodes, the first for queries and the second for documents. The representation of the input data is done through one-hot-encode, the terms are indexed by a value Long-Int, both queries and documents go through a layer of type embedding in which one-hot is converted into 100 dimensions that are used as input for two layers followed by two LSTM [7], a recurring layer capable of memorizing important information and forgetting the less. Then a fully connected layer and finally the query data and documents are aggregated through the Hadamard Product, ending with 3 fully connected layers, this architecture was inspired by [19].

The idea of this architecture is to try to merge a distributed and local model in a single architecture through the LSTM layers, memorizing the relationship that each term has to each other. As an optimization function, Adam [12] was used with a learning rate of $5E-5$ and L2 regularization. The loss function used was MSE calculated trying to predict whether a query is given and a document is positive or negative.

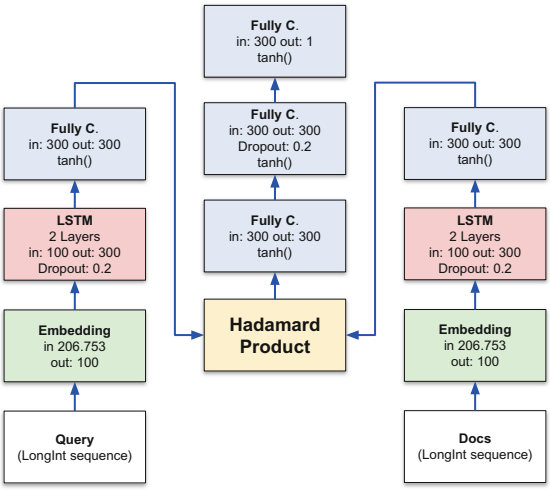

Fig. 2. Architecture Dual Embedding LSTM: Each box represents a layer with the information contained in *in* the input dimension and *out* the output dimension, in some cases we have activation layer like *tanh* and regularization *dropout*.

4 Data Characteristics

Training and evaluation use data from LExR collection [17], a public collection extracted from the *Plataforma Lattes* containing metadata from 10,942,014 publications among 206,697 candidates with title, keywords and some cases the summary. This set also includes 235 queries suggested by 513 experts who made judgments of relevance to themselves and some others, reaching 1,635 relevant judgments in all.

The keywords describe clearly and objectively the main issues that permeate the documents [24]. In this way, to generate the queries for our model, we extracted the keywords of all articles, totaling 1,876,279 valid queries. With these queries,we use the configuration mentioned in Subsect. 3.1 applied to a search server Elasticsearch [8], using the LM Dirichlet similarity function, reducing all lowercase terms and removing stop words in Portuguese and English. We then extracted up 20 documents for each query, totaling 20,641,359 triple of document, query and score.

For negative documents we use the technique of Subsect. 3.1. For each of the 1,876,279 queries, the most distant candidates were found after the reduction of dimension using the encoder, finally, we extracted the documents of the candidates until totaling the same amount of positive documents for each query.

2,456,446 terms were extracted from the documents, so we removed the terms with less than 20 uses due to the reduced size of the data and to collaborate with the model that requires a lot of training data, and may not create a good representation of the terms with few samples. After the reduction we have 206,753 terms that will be used in indexing queries and documents that will serve as input in the Dual Embedding LSTM model.

A summary of pre-processed data:

- 206.753 Terms
- 1.876.279 Queries
- 8.428.270 Documents used
- 41.282.718 Triples of Query + Document + Score (including half positive and half negative)

With data generated, the next step includes indexing documents and queries, transforming each term into its previously determined index. The data are separated into 80% for training and other 20% for validation and the 1,635 relevance judgements of LExR to perform the model test.

The training was carried out with size 300 batches, that is, 300 triples of query, document and score are inserted in the model. The architecture was developed in Pytorch and we used the Google Colab platform to train the model for 5 epochs, taking about 402 min to train and validate in each iteration.

5 Results

Applying the proposed model to the transformed data, we obtained the results from Table 1, which shows the evolution of the model according to each epoch, the accuracy increases until epoch 3, after which the model suffers from *overfitting* when the model starts to decorate the data and stops generalizing to unseen data . To calculate the *nDCG@10* we performed the test set queries in the model present Subsect. 3.1 returning 2000 documents for each query , that are finally re-ranked using the Dual Embedding LSTM. This new score is grouped by candidate and added, generating a final score for each. It is then compared with the *LExR* template as seen in Table 1.

The performance comparison between the proposals can be seen in Table 2 where we have the best models trained with negative documents randomly sampled and selected from the deep autoencoder as well as the generated baselines by LM Dirichlet and presented by [16].

Table 1. Evolution of the model by epoch

	Random		Autoencoder	
Epoch	nDCG@10	Accuracy	nDCG@10	Accuracy
1	0.159	0.725	0.167	0.6973
2	0.170	0.761	0.169	0.8737
3	0.169	0.782	**0.184**	**0.8894**
4	0.168	0.797	0.176	0.8894
5	0.166	0.808	0.174	0.8892

Table 2. Performance between different models.

Method	nDCG@10
Inf. Theoretic ρKL [16]	0.135
Inf. Theoretic ρH - ψDC [16]	0.146
Inf. Theoretic ρH - ψSDC [16]	0.164
LM Dirichlet	0.178
Dual Emb. LSTM + Random	**0.170**
Dual Emb. LSTM + Autoencoder	**0.184**

6 Conclusions

An improvement over baselines can be seen, indicating that it is possible to train a weak supervised model and obtain good results. The methodology of selecting documents with negative relevance with deep autoencoder for queries combined with the weak supervision generated by the language model for selecting documents with positive relevance and the Dual Embedding LSTM model to reclassify candidates surpassed the other techniques.

The next steps include verifying the model's performance using the new embedding language-agnostic BERT [8] to select new documents with positive and negative correlation given a query. Other works include the elaboration of a pair-wise architecture where the model receives two documents and the query, returning positive if the first document is more relevant, testing variations on the Dual Embedding LSTM and applying other statistical techniques to evaluate the models.

References

1. Balog, K., Bogers, T., Azzopardi, L., De Rijke, M., Van Den Bosch, A.: Broad expertise retrieval in sparse data environments. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 551–558. ACM (2007)

[8] LaBSE Language-agnostic BERT Sentence Embedding: https://dblp.uni-trier.de/.

2. Balog, K., et al.: Expertise retrieval. Found. Trends® Inf. Retrieval **6**(2–3), 127–256 (2012)
3. Berendsen, R., De Rijke, M., Balog, K., Bogers, T., Van Den Bosch, A.: On the assessment of expertise profiles. J. Am. Soc. Inf. Sci. Technol. **64**(10), 2024–2044 (2013)
4. Chi, M.T., Glaser, R., Farr, M.J.: The Nature of Expertise. Psychology Press, London (2014)
5. Dehghani, M., Zamani, H., Severyn, A., Kamps, J., Croft, W.B.: Neural ranking models with weak supervision. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 65–74. ACM (2017)
6. Dias, T.M.R., Moita, G.F.: A method for the identification of collaboration in large scientific databases. Em Questão **21**(2), 140–161 (2015)
7. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: continual prediction with LSTM. In: 9th International Conference on Artificial Neural Networks (ICANN1999). IET (1999)
8. Gormley, C., Tong, Z.: Elasticsearch: the Definitive Guide: a Distributed Real-time Search and Analytics Engine. O'Reilly Media, Inc., Sebastopol (2015)
9. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
10. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D.S.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1. pp. 541–550. Association for Computational Linguistics (2011)
11. Husain, O., Salim, N., Alias, R.A., Abdelsalam, S., Hassan, A.: Expert finding systems: a systematic review. Appl. Sci. **9**(20), 4250 (2019)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
13. Lane, J.: Let's make science metrics more scientific. Nature **464**(7288), 488 (2010)
14. LeCun, Y., Bengio, Y., Hinton, G.: Deep Learn. Nat. **521**(7553), 436–444 (2015)
15. Lin, S., Hong, W., Wang, D., Li, T.: A survey on expert finding techniques. J. Intell. Inf. Syst. **49**(2), 255–279 (2017)
16. Mangaravite, V., Santos, R.L.: On information-theoretic document-person associations for expert search in academia. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 925–928. ACM (2016)
17. Mangaravite, V., Santos, R.L., Ribeiro, I.S., Gonçalves, M.A., Laender, A.H.: The lexr collection for expertise retrieval in academia. In: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 721–724. ACM (2016)
18. Mitra, B., Craswell, N., et al.: An introduction to neural information retrieval. Found. Trends® Inf. Retrieval **13**(1), 1–126 (2018)
19. Mitra, B., Diaz, F., Craswell, N.: Learning to match using local and distributed representations of text for web search. In: Proceedings of the 26th International Conference on World Wide Web, pp. 1291–1299. International World Wide Web Conferences Steering Committee (2017)
20. Robertson, S., et al.: The probabilistic relevance framework: Bm25 and beyond. Found. Trends® iInf. Retrieval **3**(4), 333–389 (2009)
21. Rosenblatt, F.: The Perceptron, a Perceiving and Recognizing Automaton Project Para. Cornell Aeronautical Laboratory, New York (1957)

22. Salakhutdinov, R., Hinton, G.: Semant. Hash. RBM **500**(3), 500 (2007)
23. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 990–998. ACM (2008)
24. Yi, S., Choi, J.: The organization of scientific knowledge: the structural characteristics of keyword networks. Scientometrics **90**(3), 1015–1026 (2012)
25. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: ACM SIGIR Forum. vol. 51, pp. 268–276. ACM (2017)