Learning Job Titles Similarity from Noisy Skill Labels

Rabih Zbib, Lucas Alvarez Lacasa, Federico Retyk, Rus Poves, Juan Aizpuru,
Hermenegildo Fabregat, Vaidotas Šimkus, Emilia García-Casademont

Avature Machine Learning
machinelearning@avature.net

Problem Definition

- Measure the Semantic Similarity of Job Titles
  - Important component for measuring relevance between Jobs and Candidates
- Typical Approach is to train a Siamese network, but requires labeled data in large quantities
- Goal: Train Semantic model for Job Titles without manually labeled data

Approach

Noisy Labels Data:

- **Data Samples:** Job titles and associated skills extracted from Job Descriptions and Anonymized Resumes.
  Skills are 'noisy', i.e. extracted with simple string matching. True skills might be missed and False skills might be extracted
  Jobs with shared job title are combined, and skills merged to form samples: $(j, s_j^+ = \{s_1 : m_1, s_2 : m_2, s_{n_j} : m_{nj}\})$

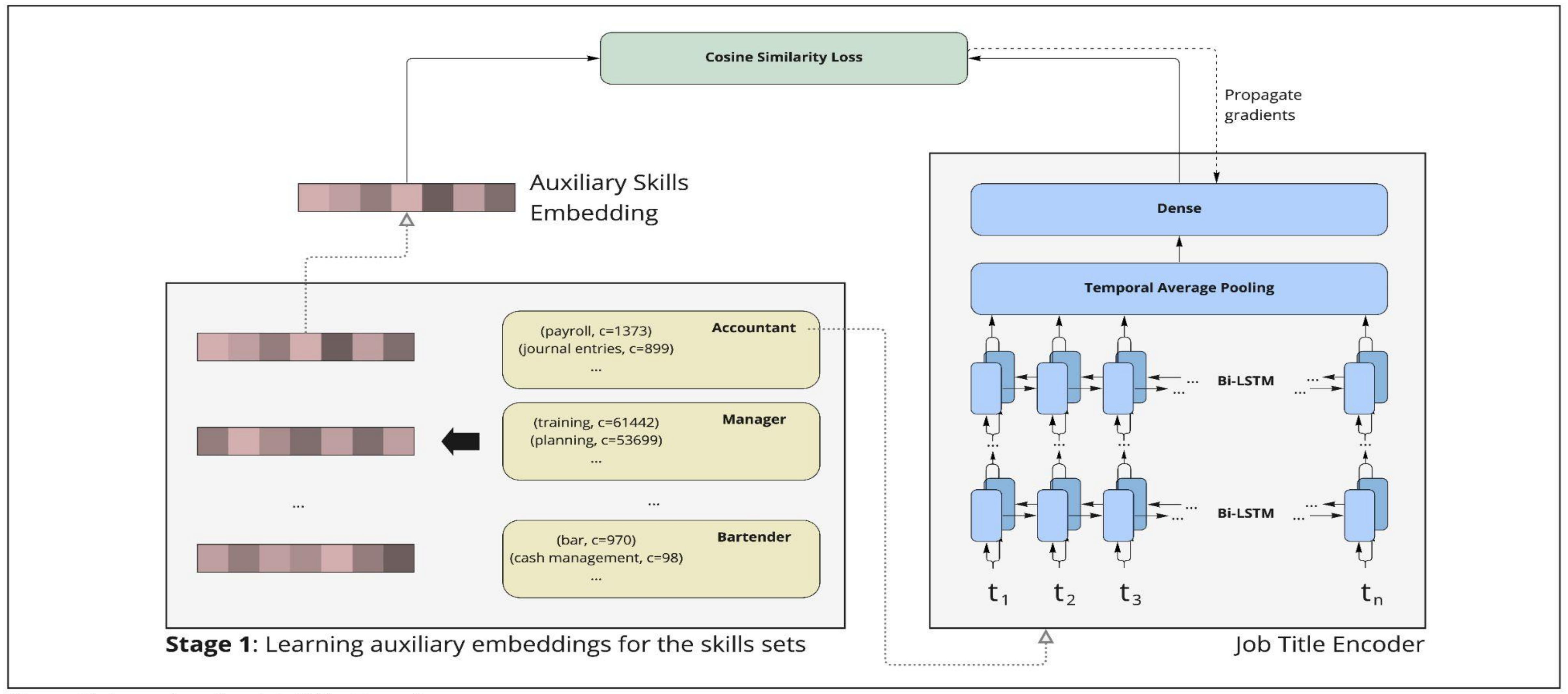
Two-Stage Approach:

- **Stage 1:** Learning Auxiliary Representation from Noisy Skills. Use the skills and counts to learn a **word2vec** representation of the job $\mathbf{e}_j$
- **Stage 2:** Training a Job Title Encoder $\eta : j \to \eta(j)$ with an RNN-based or BERT-based architecture to encode a job title to its representation. The encoder is trained to minimize the distance between the encoded job title $\eta(j)$ and the auxiliary representation $\mathbf{e}_j$

Alternative Training Procedure: Negative Sampling. Proposed by Decorte et al, 2021†.
Train the job encoder (BERT-based) to predict whether an individual skill belongs to a job.

†Decorte, J.J., Hautte, J.V., Demeester, T., Develder, C.: JobBERT: Understanding Job Titles through Skills. In: FEAST, ECML-PKDD 2021 Workshop (2021)

Model Training



Stage 1: Learning auxiliary embeddings for the skills sets

Stage 2: Learning the Job Title Encoder

Training Data

- **Skills Data Set**: 5,600 skills
- **Training Data**:
  - Raw Training Data Set: 44 million samples
  - Meged Training Data Set: 8 million samples

Text Ranking Experiments

Task:

- Input: A query Job Title
- Output: Set of Corpus Job Titles ranked by relevance
- The encoding of Query Job Title $\eta(j)$ is first computed, then Corpus Job Titles are ranked by their cosine similarity to $\eta(j)$

Test Data:

- 104 Query Job Titles
- 2,724 Corpus Job Titles
- Manually labeled for relevant (Adjudication of 2 independent annotations, 86% agreement)

Method		MAP	P@5	P@20
Text-based Retrieval				
Model	**Training Method**			
Okapi BM25	Trained on $\mathcal{D}_{\text{merged}}$	0.2754	0.5067	0.3062
BERT	None (no fine-tuning)	0.1556	0.3124	0.1871
BiLSTM	Negative Sampling	0.6428	0.7581	0.5376
BERT	Negative Sampling	0.6011	0.7238	0.5152
BiLSTM	Job Similarity Training	0.6814	0.7790	0.5781
BERT	Job Similarity Training	**0.7077**	**0.7829**	**0.5929**
Skill-based Retrieval				
TF-IDF (Noisy Test Skills)		0.3319	0.5481	0.3135
Doc2vec (Noisy Test Skills)		0.1031	0.1675	0.1204
TF-IDF (Gold Standard Test Skills)		0.7880	0.8376	0.6668
Doc2vec (Gold Standard Test Skills)		0.7126	0.7446	0.5921

MAP=Mean Average Precision. P@5=Precision at top 5. P@20=Precision at top 20.

Job Title Normalization Experiments

Task: Map an input job title to one in a set of normalized titles

Data (From Decorte et al, 2021):

- 15,463 raw job titles
- 2,675 normalized job titles from ESCO occupations corpus

Model	Training Method	MRR	P@5	P@10
BERT	Decorte et al. [1]	0.3092	0.3865	0.4604
BiLSTM	Job Similarity Training	0.3007	0.3955	0.4760
BERT	Job Similarity Training	**0.3414**	0.4595	0.5400

MRR=Mean Reciprocal Rank. P@5=Precision at top 5. P@10=Precision at top 10.

Avature