NEMO: Next Career Move Prediction with Contextual Embedding

Presenter: Jaewon Yang
Labor Flow

- Labor flow: How people change jobs
- *Why do we care?*
  - Match supply with demand
  - Circulate knowledge and drive innovation
- Large-scale studies possible with web (e.g., LinkedIn)
- We focus on predicting individual’s career moves on large scale
State of the Arts

- **Macro-level**
  - Employer-to-employer flows [Bjelland+2011]
  - Labor flow networks [Guerrero+2013]

- **Micro-level**
  - How scientists move and transform careers [Deville+2014]

- **Job Recommendation**
  - Model matching users and job postings [Paparrizos+2011]
Next Career Move Prediction

- **Given**: the sequence of positions (title / company) of users up to time $T$, users’ skill sets, education and locations

- **Predict**: a user’s next position (title/company) after $T$
Design Objectives

- **Goal-1:** *using profile attributes*
  - Career moves reflect profile info (*skills*, *education*, *location*)
  - Helps for people with short career history

- **Challenges**
  - Single-valued (*final education*), Multi-valued (*skill set*)
Design Objectives

- **Goal-2: modeling position sequence**
  - Career moves reflect the *history* of one’s past career path
  - Rare to switch to an entirely new field

- **Challenges**
  - Not enough to consider only current position
  - Model the *entire sequence* of job positions
Roadmap

- Motivations
- Proposed Solutions – NEMO
- Empirical Evaluations
- Conclusions
An Encoder-Decoder Architecture

- **Encoder**: maps profile contexts to a fixed-length vector (profile context vector)
- **Decoder**: maps profile context vector to sequence of positions
Encoding the profile contexts

- Learning **embeddings** for skills $s$, schools $h$ and locations $r$

- Aggregate all the profile contexts
  - Max-pooling on all the skills
    \[ s_u^u = \max(s_1^u, s_2^u, \ldots, s_m^u) \]
  - Concatenate skill with school/location, feed to a one-layer neural network
    \[ v_u = \tanh \left( W_v [s^u, h^u, r^u]^T + b_v \right) \]
Decoding with LSTM

- Long short-term Memory Network (LSTM)

- Latent state vector
  - Initialized by the profile context vector
  - Generates position (company / title), updated by LSTM

- Company/title independent given latent state
Learning and Prediction

- **Learning** - Maximizing the log probability of the observed career path
  
  \[- \text{Individual User’s Career Path}\]

\[
\log p(J^u | S^u, h^u, r^u) = \sum_{t=1}^{T} \log p(J^u_t | S^u, h^u, r^u)
\]

\[
= \sum_{t=1}^{T} \left[ \log p_t(c^u_t | S^u, h^u, r^u, c^u_{t'} < t, l^u_{t'} < t) \right]
+ \log p_t(l^u_t | S^u, h^u, r^u, c^u_{t'} < t, l^u_{t'} < t) \]

- **Prediction** – Given state vector, predict title, predict company
Roadmap

- Motivations
- Proposed Solutions – NEMO
- Empirical Evaluations
- Conclusions
Datasets

- Construct two datasets from LinkedIn
  - *Computer*: comp. software, internet, IT, etc
  - *Finance*: banking, investment, etc
- **Task**: predict next company/title after 12/01/2015, given position sequence up to 09/01/2015
# Mean Percentile Rank Comparison

Smaller is better

<table>
<thead>
<tr>
<th></th>
<th>Computer</th>
<th>Finance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Company</td>
<td>Title</td>
</tr>
<tr>
<td>Profile Context</td>
<td>0.0512</td>
<td>0.0286</td>
</tr>
<tr>
<td>MC (no profile, last position only)</td>
<td>0.0542</td>
<td>0.0277</td>
</tr>
<tr>
<td>LSTM (No profile)</td>
<td>0.0432</td>
<td>0.0225</td>
</tr>
<tr>
<td>NEMO</td>
<td>0.0299</td>
<td>0.0182</td>
</tr>
</tbody>
</table>

LSTM > MC: entire sequence > last pos.

NEMO is best – profile context + career path
Stratification by Seniority

- NEMO > LSTM (x < 5): Profile context helps for juniors
## Case Study

<table>
<thead>
<tr>
<th>Previous position</th>
<th>Ground-truth next position</th>
<th>Top 5 predicted Company</th>
<th>Top 5 predicted Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior project manager @ Fidelity Investments</td>
<td>Project Manager @ Southwest Airlines</td>
<td>Fidelity Investments American Airlines Southwest Airlines Epsilon Bank of America</td>
<td>Senior Project Manager Project Manager Technical Project Manager Senior Technical Project Manager Program Manager</td>
</tr>
<tr>
<td>Worked at airline before</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Architect/Tech Lead @ Bureau of Labor Statistics</td>
<td>Consultant @ United States Patent and Trademark Office (USPTO)</td>
<td>Fannie Mae USPTO FINRA Lockheed Martin Freddie Mac</td>
<td>Technical Lead Senior Software Engineer Consultant Senior Consultant Solutions Architect</td>
</tr>
<tr>
<td>Worked at USPTO before</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Sampling Career Path

**User 1**
SF Bay Area  
CMU graduate  
Skills: ML, DM, AI, Algorithms  
1st job: ML Engineer @ Google

ML engineer @ WhatsApp  
ML engineer @ Uber  
Data Scientist @ Facebook

Both have a rising career trajectory

**User 2**
Greater New York Area  
Harvard Business School  
Skills: Financial services, investments

Investment banker @ Citi  
Technology Strategist @ Citi  
Relationship Manager @ Citi

Cold-start Case  

Engineer Lead @ LinkedIn  

VP brokerage @ JPMorgan Chase  

VP @ Morgan Stanley
Conclusions

- **Problem Def**: next career move prediction

- **Design Objectives**: profile context + position sequence

- **Solutions**: NEMO – contextual LSTM

- **Results**:
  - Profile + Position sequence = better prediction
  - Insights from career path sampling
Thank you!
Varying Embedding Dimension

Diminishing return in all the models
Stratification by Popularity

NEMO performs significantly better when company/title is rare