John Hopkins University

Center for Language and Speech Processing

—

Dushyanta Dhyani
Karan Grover

Group Introduction

Areas of research - Machine Translation, Probabilistic model of Linguistic structure, Representation learning, NLP for Social Media, Computational Semantics

Members:

- 16 Professors
- 1 Research Scientist
- 3 Post-docs
- 56 PhD students

Publications in 2016 (over 50 in total) :

- ACL - 6
- EMNLP - 6
- NAACL - 4

Faculty and Research focus

Raman Arora - Representation learning, multi-view learning, deep learning and their applications to speech and language processing.

Mark Dredze - Graphical models, semi-supervised learning, information extraction, large-scale learning and speech processing

Jason Eisner - Probabilistic language processing, combinatorial algorithms

Philipp Koehn - Statistical Machine Translation (book published with the same name)

Benjamin Van Durme - Knowledge acquisition, Linguistics Semantics

Fluency Detection on Communication Networks

Tom Lippincott and Benjamin Van Durme; 2016 EMNLP

- The paper presents a model to leverage the structure of Twitter data (tweets and “following“ relationship between users) for detecting fluency of a user in a set of languages.

- For fluency detection, traditional language identification (LID) models such as the langId tool (Lui and Baldwin, 2012) when used individually do not perform well enough on tweets because of short length of tweets and the language usage being idiosyncratic.

http://aclweb.org/anthology/D/D16/D16-1107.pdf

Structure-Aware Fluency Model

The model uses the communication network structure (Tweets and “following” relationship between users) in conjunction with the LID model to determine the fluency in a set of languages.

Notations:

- $A_{1:T}$ - Actors or Users who send/receive messages
- $M_{1:U}$ - Messages or Tweets
- $F(a_i)$ - Binary vector indicating which languages we believe actor a_i is fluent in
- $L(m_i)$ - One-on binary vector indicating which language we believe message m_i is written in.
- $P(m_i)$ - Set of actors participating in message m_i. For twitter, the participant set is the set of user and the followers of that user.
- $LID(m_i)$ - Real vector representing the probability of message m_i being in each language, according to the LID system

Structure-Aware Fluency Model (continued)

The model is specified as Integer Linear Programming (ILP) model with the following structural constraints:

- Each message has a single language assignment

$$\sum L(m_i) = 1$$

- All actors participating in a given message are fluent in its language:

$$\forall a \in P(m_i), L(m_i) \times F(a) = 1$$

Structure-Aware Fluency Model (continued)

Objective function of the model has two component:

1. **Language Fit** - encourages the model to assign each message a language that has high probability according the the LID system:

$$\mathrm{LF} = \sum_{m \in M} L(m) \times LID(m)$$

2. **Structure Fit** - minimizes the cardinality of the actors' fluency sets (subject to the structural constraints), and thus avoids the trivial solution where each actor is fluent in all languages:

$$\mathrm{SF} = -\sum_{a \in A} \sum F(a)$$

Complete Objective Function:

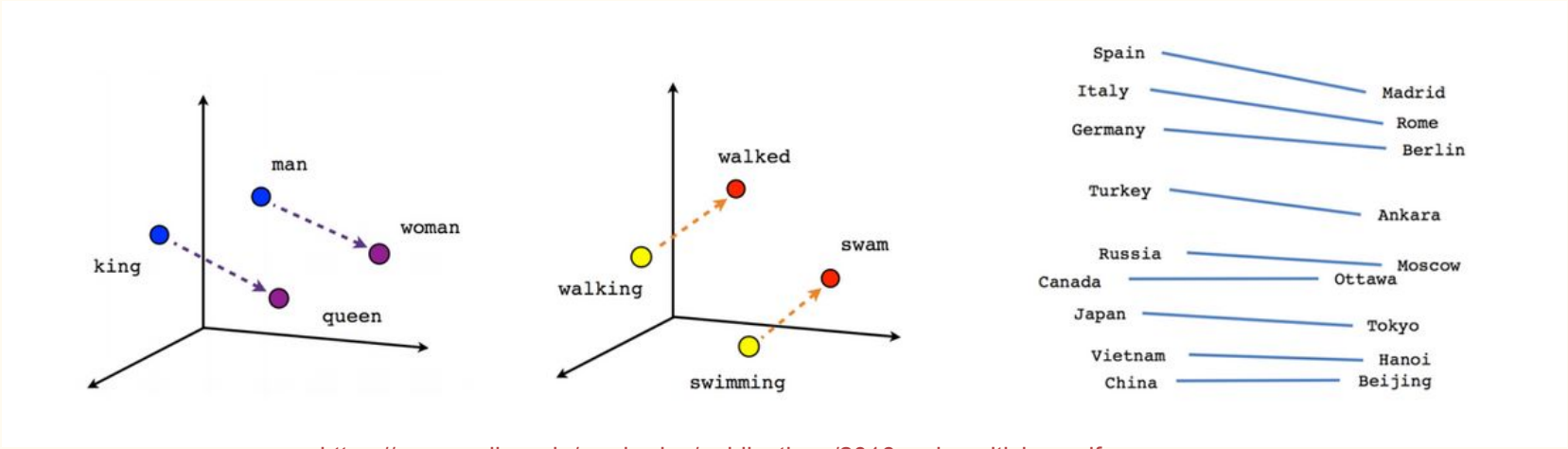
$$\mathrm{LW} \times \mathrm{LF} + (1.0 - \mathrm{LW}) \times \mathrm{SF}$$

Where LW is the empirically determined Language weight

Learning Multiview Embeddings of Twitter Users

Adrian Benton, Raman Arora, Mark Dredze; ACL 2016

- A lot of work has been done for learning compact and expressive representations of words.
- Word2vec (Mikolov et al.) demonstrated the power of these representations.



https://www.cs.jhu.edu/~mdredze/publications/2016_acl_multiview.pdf

Image Source : https://www.tensorflow.org/versions/r0.11/tutorials/word2vec/index.html

Learning Representations of Users/Entities

- Can be used to learn important characteristics of users/entities/participants.
- A simple approach can be to train word2vec on the tweets of users to model their latent characteristics.
- As the authors suggest, there are other ways **(views)** to characterize a user
  - Users the user follows
  - Users the user is friends with
  - Users the user mentions in their tweets
- Each view can also vary in the modalities i.e. text, image, hyperlinks, etc. (the authors currently restrict their approach only to text)

Some Potential Applications of User Embeddings

- **User Engagement -** Predicting what topics a user would tweet about.
- **Friend Recommendation**
- **Predicting Demographic Characteristics -** Gender , Political Affiliation , etc.

However, each of the above application depends on different proportions of the previously mentioned views. Thus simply combining (i.e. concatenation of representation vectors) might not produce good results.

How to Use (and obtain) Views?

- For a particular user in a particular view , treat set of all tweets as a single document.
- For each user in a given view obtain the following representations
  - **BOW (Bag of Words)**
  - **PCA -** PCA on each of the views and also a combination of each of them
  - **Word2Vec -** Helps to learn non linear representations of the users as compared to the linear representations of the previous approaches.
  - **NetSim** - Using all the users as a vocabulary in a bag of words model.
  - **NetSim-PCA -** Performing PCA on the above NetSim Model.

Combining Views

- Use Generalized Canonical Correlation Analysis (GCCA) to combine multiple views into single embedding.
- GCCA is a multi-variate variant of Principal Component Analysis (PCA)
- Since views may have different impacts on domain problem, we take a weighted combination of views in GCCA.
- This leads to a domain specific transformation of multiple views into a single dense representation.
- These representations can then be used for recommendation tasks (using metrics like cosine similarity) or classification task (by feeding them to ML Classifiers)

THANK YOU!