A Model for Dynamically Embedding Twitter Users for User Profiling

Daniel Dicken

1. Introduction

In the task of dynamic user profiling, we seek to describe a users interests and characteristics within a system over a period of time, dynamically updating and classifying this information as our dynamic time interval progresses. The information we infer about users in a system can be fed into downstream applications that can take advantage of this information allowing application developers to make better products.

As a forewarning, this project is not nearly complete. This is due to a combination of not getting started early enough and not correctly quantifying the amount of effort needed to implement the DUWE.

2. Project Description

This project seeks to replicate the work done by Liang et all. We want to replicate their model which allows us to generate user embeddings of twitter users over time. As per [1], “given a set of users and a stream of short text generated by them in Twitter, infer both user and word semantic representations over time, and dynamically identify a set of top-K relevant and diversified keywords to profile each of the users.” This allows applications taking advantage of the model to learn information and cater directly to a users profile.

This model works by creating embeddings of users and words within the same embedding space. Given a set of user and word embeddings in a space, the query for retrieving the top K relevant words for a user then falls out; you simply find the nearest word embeddings at a time *t* for a given user embedding at the same time *t* (this process is referred to as the Streaming Keyword Diversification Model). Thus, the important and difficult part of this problem is creating the embeddings.

Typically, we could do a simple calculation given word counts to create these embeddings. However, the problem we are trying to solve is not in closed form, so we can not compute the solution with a closed computation. The embeddings are created in several steps:

1. Collect counts of words, word pair frequencies, and inverse pair frequencies (the frequency of pairs not appearing) at every time step
2. Implement the computations needed to run the dynamic user word embedding model
3. Write the DUWE (Dynamic USer Word Embedding) model to take advantage of these calculations to create embeddings

After these steps have been completed, we can write the simple Streaming Keyword Diversification Mode algorithm that takes advantage of the embeddings at each time step *t*. After implementation is completed, the goal was to train the model on a twitter user dataset and test against the same metrics used in the paper.

3. Work Accomplished

As stated earlier, this project is nowhere near complete. This is due to a combination of not getting started early enough and not correctly quantifying the amount of effort needed to implement the DUWE. However, some work can be described and quantified.

3.1 Gathering Data

The first task was to gather training and test sets of data. This was accomplished by reaching out to the publisher of the article I was attempting to replicate, Shangsong Liang. This process took awhile, but eventually I was able to get ahold of the training data set after Liang directed me towards people that could give me access. So, I was able to obtain the large twitter user data set that was used by Liang in his article.

I was not able to get a hold of the annotated test data set via Liang. Liang said that a Chinese company was contracted to create the annotated test data and he did not have the rights to release this data to me. The test data Liang used contained hundreds of Twitter users annotated data. I did not have the capacity to do this annotation, so I decided to take the following approach. I chose ten users data to annotate for testing. I considered time steps of one day within the data, annotating up to the top 10 diverse words for each time step given the new document set. These annotations were completely subjective, but it was the best solution I could come up with.

Liang also said that I could send him my model upon completion and he could test it against the real test data, but I was not able to complete the project, so this interaction never happened.

3.2 Implementing DUWE

As outlined in section 2, the DUWE is the complicated part of this project. Of the outlined steps, progress was made on two. First, I made progress towards the counting of word frequency in documents. The counting the did not happen yet was for word pairs and inverse word pairs.

Progress was also made on step 2, implementing the required calculations for DUWE. I implemented most of the needed calculations from section 4.2 of the paper. The task of implementing these calculations was much harder for me than anticipated. All of the different variables and equations working with each other lead to lots of confusions for myself, which created a very slow development process.

No progress was made towards step 3 of creating the DUWE.

3.3 Implementing Streaming Diversification Model

No progress was made towards this portion of the model.

3.4 Testing the Model

For testing, the metrics I was planning on testing against were RQ3 i.e. the quality of semantic representations outlined in section 7.3 and RQ4 i.e. dynamic representation against the test data.

4. Conclusion

In this project, I hoped to solve the problem of user profiling over time given twitter user tweets. Through this project, I have come to understand the skip-gram-esque embedding model that is implemented in the paper. Unfortunately, due to poor planning, most of my project goals remain unobtained at this point.

References

1. Liang S, Zhang X, Ren Z, Kanoulas E (2018) Dynamic Embeddings for User Profiling in Twitter. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD ’18. Available: http://dx.doi.org/10.1145/3219819.3220043.