Proceedings of the Fourteenth International AAAI Conference on Web and Social Media (ICWSM 2020)

Learn2Link: Linking the Social and Academic Profiles of Researchers

Asmelash Teka Hadgu
Leibniz University Hannover
asmelashteka@acm.org

Jayanth Kumar Reddy Gundam
Leibniz University Hannover
jayanth.gundam@gmail.com

Abstract

People have presence across different information networks on the social web. The problem of user identity linking, is the task of establishing a connection between accounts of the same user across different networks. Solving this problem is useful for: personalized recommendations, cross platform data enrichment and verifying online information among others. In this paper, we propose a deep learning based approach that jointly models heterogeneous data: text content, network structure as well as profile names and images, in order to solve the user identity linking problem. We perform experiments on a real world problem of connecting the social profile (Twitter) and academic profile (DBLP) of researchers. Experimental results show that our joint model achieves a 97% F1 score outperforming state-of-the-art results that consider profile, content or network features only.

Introduction

With increasing content in user-generated Web, the use of online social media has changed human experience and the way people communicate and interact with each other. A study by (Fox and Jones 2013) found that an increasing number of adults and more than 93% of teens have been online. Almost all of the users have been using social media, mainly for communication purposes. This communication is not only limited to personal interactions, but also helps sharing knowledge and maintaining professional contacts. This is particularly evident in the scientific community including researchers, publishers and readers to participate in scientific communication processes supplementing conventional bibliometric approaches, amplifying the scientific impact of publications.

The increasing participation of users on social media has given everyone a platform to have their say but at the same time brought some issues such as account impersonation on social networks (De, Bogart, and Collins 2012). It is at times difficult to tell opinions of experts on a subject matter from those that echo popular opinions based on hunch on topics ranging from climate change to consequences of artificial intelligence in society. At its core, this problem calls for methods that link user identities across social networks.

Current advances in deep learning now enable researchers to solve many supervised learning tasks that previously required tedious manual feature engineering without having to explicitly construct features manually. This is partly due to feature representation learning methods that take raw signals such as images, text and network structure and provide compact representations that can be used in downstream machine-learning tasks. Examples of such representation learning methods include word2vec (Mikolov et al. 2013) and node2vec (Grover and Leskovec 2016) for text and networks respectively. These representations (embeddings) have been successfully used in computer vision as in (Zagoruyko and Komodakis 2015) where the authors learn directly from image data (i.e., without resorting to manually-designed features) a general similarity function for comparing image patches. Similarly in (Lu and Li 2013) the authors proposed a deep learning model to matching tasks in natural language processing problems such as question answering.

In our work we leverage these advances in representation learning that use deep neural networks to harness heterogeneous data, i.e., profile level features such as names and images, content features and network structures in order to match users across social networks. To the best of our knowledge (Shu et al. 2017), this work is the first to present experimental results that leverage all three types of heterogeneous data (profile, content and network) with three modalities: text, image and network in a supervised learning scheme to perform user identity linking across social networks. Our contributions are:

- A novel approach that leverages deep learning to jointly model profile, content and network features to solve the problem of user identity linking.
- Application on linking real-world networks: the social (Twitter) and academic (DBLP) profiles of researchers in Computer Science.
- Open source data and code [1] for reproducibility and other researchers to build upon.

[1] https://zenodo.org/record/3735448

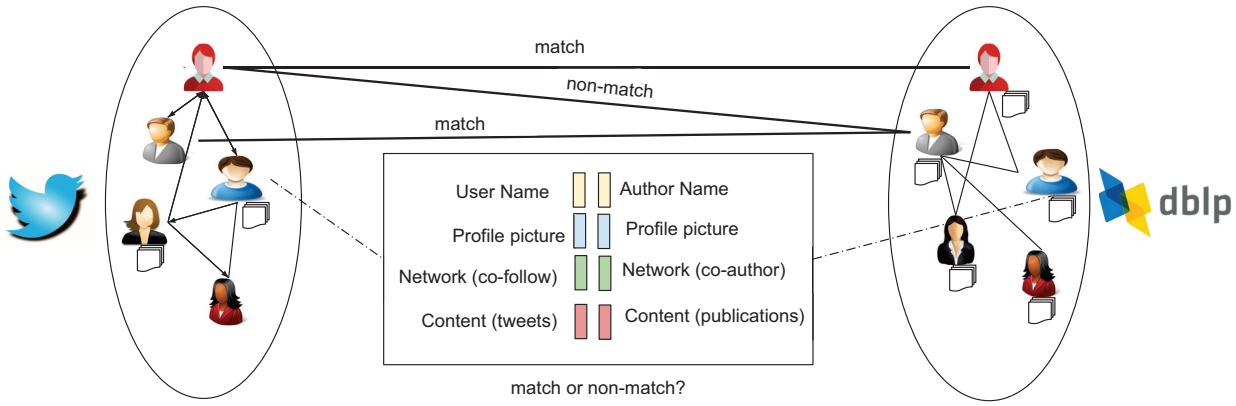

Figure 1: An overview of user identity matching across online networks. On the left is a social network (Twitter) and on the right an academic network (DBLP). From a small subset of example pairs (match, non-match), our system learns to jointly model pairwise signals (name, picture, network, content) to determine whether a candidate pair is a match or non-match.

Related Work

There has been a lot of work on the problem of user identity matching across networks. We will discuss some of the most relevant ones to our work. We refer the interested reader to (Shu et al. 2017) for an extensive literature review on the subject.

Profile Based Linking. The use of username in a user profile to match users identities across different networks has been proposed by (Perito et al. 2011). The authors used an analytical model that estimates the uniqueness of a username which can be assigned a probability. This estimate is used to construct a classifier that determines with a probability that a pair of usernames belong to the same user in two different platforms. An unsupervised approach using distance based techniques has been proposed by (Liu et al. 2013). The authors demonstrate the importance of alias-disambiguation step by experimental analysis on about.me data where they show that the rareness of a username measured by its n-gram probability is a good indicator of profiles belonging to the same person. They automatically create a labeled training data with knowledge of n-gram probability. Our approach builds on these works in that the username is indeed an important feature. Unlike these works; however, we do not explicitly construct features to match names. We learn the similarity of names by leveraging a large collection of multiple names of the same person.

Content Based Linking. Content features usually reveal a person's interests and involvement in communication with other users on a platform. These features also give us an idea of the person's area of work, e.g., their recent publications if they are a researcher, and the organization the user is affiliated with. In (Kong, Zhang, and Yu 2013), the authors use the hint that different users have different use of words in their posts and converting these posts into bag of words vector weighted by TF-IDF and comparing the vectors using inner product method and cosine similarity of the vector to find similar accounts. (Goga et al. 2013) proposed a probabilistic model to correlate accounts using textual data. The authors considered data from yelp and flickr photo descriptions to connect to twitter accounts by building a unigram probability distribution. To measure the similarity, the probabilities of each word in yelp and flickr accounts are accumulated from the language model of the twitter accounts. We build on the observations of these previous works to incorporate content information in user identity matching. Unlike the previous works we do not create explicit word features for matching. Instead, we collect data on similar content posted across networks and learn a content matching model.

Network Based Linking. The use of network features to match user identities has been shown by (Man et al. 2016; Liu et al. 2016). In (Man et al. 2016) the authors proposed entity linking by using network based embeddings to match users across different platforms. Given two networks, the authors first construct vector representations in low dimensional space as embeddings which preserve the structural information leveraging previously observed anchor links as supervised information. Then these embeddings are matched using linear mapping for linear relationships and multilayer perceptron model to capture non-linear relationships. Use of friends list from two different networks of user and matching the users by overlap of friends has been proposed by (Labitzke, Taranu, and Hartenstein 2011). The authors used the distinction distance metric which compares the overlap of matches with more number of candidates in the intersection with the next lower overlap, where higher distinction distance indicates the profiles are matched with higher probability. In our work, we use ideas from (Man et al. 2016) to build network embeddings on the separate networks and use these representations to perform user identity matching. Our joint model incorporates profile and content level features besides network features for a holistic view.

Combined Approaches. A semi-supervised approach to link entities using user names and content features was done by (Iofciu et al. 2011). The authors make use of tags present in the content of user profiles and model each profile as a vector with each TF-IDF value of a tag as a dimension. Matching is determined using cosine similarity between vectors. To match user names, string similarity measures such as Jaccard and Levenshtein distances are used. Finally a mixture model using normalized score based on each of the feature is used to combine the similarity score. Another work that uses a combined approach is (Kong, Zhang, and Yu 2013), where anchor link prediction across heterogeneous multiple social accounts is proposed. The authors use the insight that different users have different use of words in their posts and converting these posts into bag of words vector weighted by TF-IDF and comparing the vectors using inner product method and cosine similarity of the vector helps in finding the similar accounts. For the purpose of network information, the authors make use of spatial and temporal data of the users to find the same user accounts. Then a classifier is applied on these features and a multi network anchoring is used to infer links based on ranking scores of the classifier.

Problem Definition

Let us formally describe the user identity linking problem. Let u be a user identity representation on a social media site for a person P. This consists of three components: Profile, Content and Network. Profile includes, user describing features such as user name and profile picture. Network refers to the set of attributes that describe a user's social connections with other users in the network. Content consists of user generated items such as text posts, video shares etc. An online social network G is represented as a graph $G(U, E)$ where $U = \{u_1, u_2, \ldots, u_n\}$, is the set of user identities and $E \subseteq U \times U$ is the set of links in the network.

The problem of user identity linking is defined as follows: Given two online networks G^s and G^t, the task is to determine whether a pair of user identities u^s and u^t chosen from G^s and G^t respectively belong to the same person. I.e., A user identity linking procedure attempts to learn a matching function f, such that

$$f(u^s, v^t) = \begin{cases} 1: & \text{If } u^s \text{ and } u^t \text{ represent the same person;} \\ 0: & \text{Otherwise;} \end{cases} \tag{1}$$

Approach

If you ask a human surfer to check whether a user account in one network, say Twitter, is the same as a user account in another network, say DBLP, the person will look for all possible comparable clues of the pair of accounts to make an educated guess. This involves comparing: (i) the profile picture and user name on Twitter to the corresponding picture (from their homepage if it exists) and name of the user on DBLP to (ii) the content of posts on Twitter versus the titles of the articles the person published on DBLP and (iii) the network (friends and followers) a user on Twitter to the co-authors of the user on DBLP among others.

Our system mimics such a human surfer to perform the user identity matching by jointly modeling profile level (profile image and user name), content and network features, see Figure 1. We do this by leveraging representation learning (embeddings) to compare the corresponding pair-wise signals and casting the overall task as a classification problem. In the following section, we describe in more detail how we construct these representations and carry out the classification task.

Profile Representation. Most social networks identify users by their name. Some social networks also contain profile pictures. This feature is prevalent and can be used to match user identities across networks. The key idea here is to learn a good matching function through character embeddings for names and image embedding for profile pictures. We can then learn a matching function on these representations to link users using profile level features.

Network Representation. A common characteristic of social networks is the network structure among users. On a social network such as Twitter, we have following relationship. On academic networks we have citation networks and co-author networks. In our work, we learn network representations on the respective networks independently and perform a matching on these embeddings from anchor links. In this context, anchor links are observed links of user identities between the networks.

Content Representation. Users generate content on different online networks. Overlapping content on different networks can be used to establish linking the identity of users. For example, some authors tweet about their publications on Twitter. Their academic publishing network also contains the list of their publications. Sentence level representation and matching of such content gives us evidence to link the user identities across those networks.

Learning to Link from Embeddings. We perform matching at two levels. The first one takes a pairwise representation of the profile, network and content of a user from the social network and the academic network. The matching function takes as input a pair of embeddings and learns if they represent the same identity. We use a multilayer perceptron (MLP) on a subset of training data to learn this nonlinear relationship. After each representation is matched, we then take the outputs of these matching scores and feed them to a second MLP that jointly learns the user identity matching by taking the results of the profile level matching, content level matching and network level matching.

Data

In this section, we describe the datasets used in our experiments. Our work is motivated by the task of linking the social and academic profiles of researchers. We used Twitter as an instance of the social network of researchers. Twitter is a widely used platform for scholarly communication (Mahrt,

Weller, and Peters 2014). For the academic network, we used DBLP, a digital library of computer science research.

Researchers on Twitter

The dataset for researchers on Twitter builds on the work by (Hadgu and Jäschke 2014) which is publicly available on GitHub [2]. The paper describes a machine learning approach to identifying computer science researchers on Twitter. We updated the dataset to suit our work as follows:

The first requirement was to get an updated version of the dataset. Using the same approach in the paper, we updated the seed list by adding new accounts for conferences from the list of computer science conferences on Wikipedia [3] giving us 268 seed accounts. We then gathered all followers, friends and those users that retweeted these seed accounts. This gave us 187,732 candidates. We used a supervised learning approach to identify whether a Twitter user is a researcher or not. We used positive examples from DBLP (linked to their Twitter accounts) and negative examples manually verified after generating likely negative users with list based approach (Sharma et al. 2012). Finally we built a binary classification model (Hadgu and Jäschke 2014) by constructing features such as existence of some patterns like PhD, researcher, professor etc. in the user description and presence of tilde in their URL. This approach achieved an F1-score of 94% resulting in 53,830 candidate users classified as researchers.

Disambiguated authors on DBLP

The academic network of researchers contains digital libraries that index the works of researchers and links to their personal websites and blogs. The DBLP computer science bibliography is an on-line reference for open bibliographic information on computer science research. We use the snapshot from December 2018 in our experiments. We build on the work by (Kim 2018) to take a sample of the entire authors that have been verified manually by DBLP. These disambiguated users usually contain a URL, an affiliation or a number attached to the researcher name in case there are duplicates found with the same name.

Linking Users on Twitter and DBLP

Now that we have identified users for the social and academic networks, let us look at our semi-automated approach to generate ground-truth data that establishes links between the same users on both networks.

One of the most intuitive ways to establish a connection between the same user across social networks is to check if there is a hyperlink (URL) present in one network that can either directly or through a second-level URL lead us to the same URL on the other network. We use this technique as follows to link users between Twitter and DBLP.

- From the Twitter data, we look at the URLs of the user profiles and unshorten them. We do the same on the DBLP side. If we find a common URL between users across the networks, the pair is a good candidate for a match. The most useful links in this regard were those pointing to Google Scholar and ORCID as they are the most frequent on DBLP.
- A similar approach is to check for second-level URLs. In particular, from the DBLP data, we check if the homepages or blogs pointed to by the URLs for a given user contain links to Twitter.
- For users on Twitter, if they contain links to their homepages, another approach we considered to generate candidates is the use of publications that are available in both DBLP records and in the personal page URL available via Twitter.
- Finally, there are DBLP records which already contain links to Twitter. These matching pairs were also included into the list of candidates.

The candidates generated using these approaches were further checked manually to avoid duplicates, wrong links, accounts belonging to researchers' affiliation and fake accounts. We were able to generate: 1082 (0.20%) pairs through direct and indirect links, 2649 (0.48%) through publication 1751 (0.32%) on DBLP for a total of 5375 DBLP-Twitter user pairs.

Table 1 shows the break-down of users in the ground-truth dataset by the availability of different features useful for user identity linking. Names are the most common followed by network, images and content.

	Names	Images	Network	Content
DBLP users	5,375	2,508	5,308	5,375
Twitter users	5,053	3,861	4,892	1,751

Table 1: Ground-truth for user identity linking.

Candidate User Pairs Generation

An exhaustive pair-wise comparison of a user from one network to every other user on the other network is ineffective. In practice, some heuristics are used to reduce the search space significantly. The candidate generation step is used to generate for a given user from one network a small subset of the other network that contains the same user on the second network. Mathematically, the goal of the candidate generation step can be formulated as follows. Given two sets S_1 and S_2, the task is for a given element e_i in S_1 to generate a subset S_2^i of S_2 that contains an element e_j that matches e_i from S_1.

This step generates a set of matching and non-matching DBLP and Twitter user pairs that we used to train and evaluate different user identity matching models. In our implementation, we started from the manually verified ground-truth data in the previous step. This was then expanded by adding users based on overlapping names, co-author information and affiliation information.

[2] https://github.com/L3S/twitter-researcher

[3] https://en.wikipedia.org/wiki/List_of_computer_science_conferences

Methodology

In this section, we describe the details of how we materialize the outlined approach using concrete implementation methods.

Profile Representation

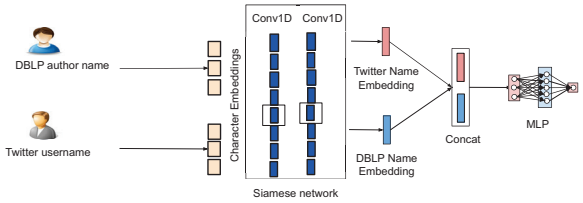

Figure 2: Siamese network on character level embeddings for user name matching.

Learning to match names. We used disambiguated users in DBLP to construct large scale name variants. 15,617 of the 84,368 disambiguated users on DBLP have at least two names, e.g., user homepages/09/11295 on DBLP has the following names ("Linda Bushnell", "Linda G. Bushnell"). From such names, we generated a pairwise combination of matching names. Similarly, we constructed non-matching pairs by forming pairs of names from two distinct users in the disambiguated users. To help our model learn a better representation, instead of taking a random pair, for a given name, we pair it with a different user that shares at least a part of their name (first name or last name) with the target name. In particular, we gathered names that shared first name or last name with the person. For example, here is a non-matching pair from our dataset: ("Borja Martínez", "Borja Sanz"). Using this technique we generated 40,289 matching and 40,891 non-matching names. The architecture we used to check similarity between a pair of names is shown in Figure 2. It is a Siamese network that uses character embeddings, passes them over a one-dimensional convolutions (Conv1D) and finally applies an MLP to determine the matching score. This is similar to Architecture-I in (Hu et al. 2014) where we used character embeddings instead of word embeddings.

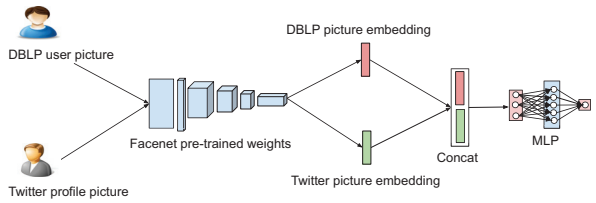

Figure 3: Transfer learning for profile picture matching.

Learning to match profile images. Profile images provide a strong signal for humans to identify and match users. In our work, each image is passed through a neural network and transformed into a vector representation. The model is trained such that it learns a mapping of the images to euclidean space so that similar images are closer to each other compared to random images. Figure 4 shows a t-SNE projection of embeddings for a pair of Twitter and profile images from researcher web pages. A researcher may have more than one profile picture obtained from links on their DBLP profile, e.g., from multiple academic pages, a blog etc. We can see that embeddings of the same user are closer together.

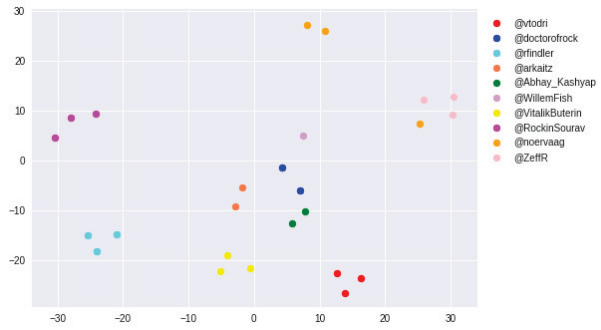

Figure 4: Profile image embeddings of Twitter and Academic (homepage or blogs etc.)

The previous embeddings for profile images were generated using the OpenFace (Amos, Ludwiczuk, and Satyanarayanan 2016) implementation which is based on FaceNet (Schroff, Kalenichenko, and Philbin 2015). FaceNet is a deep convolutional neural network model that has two parts inspired by models published in (Zeiler and Fergus 2014) and the Inception model of (Szegedy et al. 2015). The first part consists of interleaved layers of convolutions, non-linear activations, zero paddings, batch normalizations and max pooling layers. The second part inspired by the inception architecture (Szegedy et al. 2015) proposed a sparsely connected architecture using the idea that clustering sparse matrices into relatively dense sub-matrices tends to give state-of-the-art performance for matrix multiplications. The model is trained on image triplets from the Face scrub and CASIA-Webface datasets containing about 200 million images from 8 million people. The concept of triplet loss is employed to train the model, where the training data consists of three images per batch, one anchor, a positive image and a negative image and the model is trained such that the loss between anchor and positive image is smaller than the distance between the anchor and negative image at least by a margin.

We used FaceNet pretrained weights as powerful feature extractors to generate the embeddings that we can then use in our user identity matching task in a transfer learning scheme. Our architecture for profile picture matching is shown in Figure 3. Each image from profile of a user from our dataset, passes through a dlib face alignment algorithm (Kazemi and Sullivan 2014) which detects and aligns a face such that the eyes and the nose are in the same position for every image. Then this image is passed through the pre-trained weights to get a representation of an image as an embedding of length 128. These embeddings are then concatenated and

pass through an MLP to learn the matching function of profile images.

Network Representation

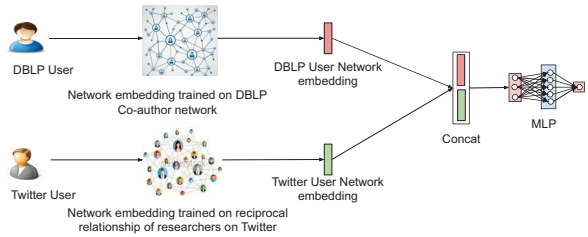

Figure 5: Network embeddings for user identity matching across networks.

A common characteristic of social networks is the network structure among users. On a social network such as Twitter, we have directed relationships, users that one follows called friends and users that follow a user called, followers. On academic networks we have citation networks and co-author networks. In our work, we learn network representations on the respective networks independently and perform a matching on these embeddings from anchor links. These are observed links of identities between the networks.

In (Pujari et al. 2015) the authors show that strong reciprocal relationships on Twitter correlate with strong reciprocal relationships on DBLP. We first form a reciprocity network of researchers on Twitter. A reciprocal relationship is one where two users follow each other. On the DBLP side, we construct the co-author network of the disambiguated users. A key component of network linking is to perform an expansion of the networks through observed anchor links (Man et al. 2016). Anchor links are users that are on the two networks and we know they match (refer to the same identity). We get these from our labeled dataset of profile matches. We extend the reciprocal network and the co-author network using these observed anchor links.

Let us take two users from our ground-truth dataset of labeled profile matches. If they have a common connection on one network, so do their counterparts on the other network. For instance, if two users are on the co-author network and our reciprocity network does not contain an edge between these users, we extend the reciprocity network to add this link. Similarly, we extend the co-author network. Extending the networks is an essential step. This is because (i) we can have missing links due to not crawled users (protected users on Twitter) and since it is not possible to observe the entire co-authorship network. We added 3509 (0.155%) new edges to the reciprocity and 106764 (29.87%) new edges to the co-author networks. After extending the networks, we learn network embeddings independently on the two networks using node2vec (Grover and Leskovec 2016). Node2vec takes as input an edge list between nodes of a graph and returns the embeddings for each node, using the skip-gram architecture for network. The architecture of our network based user identity matching is shown in Figure 5. Given a pair of users $(u1, u2)$ from Twitter and DBLP, first we generate their embeddings by passing them through the learned reciprocal and co-author embeddings. These embeddings are then concatenated and passed through an MLP to learn whether a pair of users match or not.

Content Representation

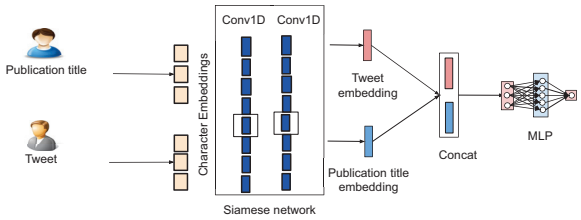

Figure 6: Siamese network on character level embeddings for content matching.

Users generate content on different online networks. Overlapping content on different networks can be used to establish linking the identity of users. For example, some authors tweet about their publications on Twitter. Their academic network on DBLP contains the list of their publication. Having a match gives us evidence to link the user identities. To learn such a pattern, we gathered tweets that have URLs that point to arxiv.org or dl.acm.org from all the tweets in our identified researchers on Twitter. We gathered 33,116 and 7,603 respectively that we combine and used for training. As a preprocessing step, we only keep English tweets as most publication titles on DBLP are in English. This is used to determine if a pair of users match based on content.

The architecture of our content matching model is shown in Figure 6. For a pair of users $(u1, u2)$, we take tweets of $u1$ and check if they match with any of the titles of the articles of $u2$ from DBLP. We use character-level sentence representation to encode the tweet quoting an article and the article title. Similar to the name matching architecture, we pass these input encodings through Conv1D and finally apply an MLP on the learned embeddings to match if a tweet refers to a publication of the same author.

A Joint Model

Given a pair of users, our unified model is an end-to-end system that takes a pair of user accounts as input and performs pair-wise comparison of the corresponding profile-picture, name, content, and network embeddings, i.e., with out explicitly hand-crafting features, as well as their aggregate to determine if the input pair represents the same user. The architecture of the system is shown in Figure 7.

We build up on current advances in character, image and network embeddings and take these as basic building blocks to come up with the whole solution. In particular, we leverage current advances in deep-learning in the following ways:

- transfer learning is used to leverage a model trained on millions of images to compute embeddings for profile pictures matching

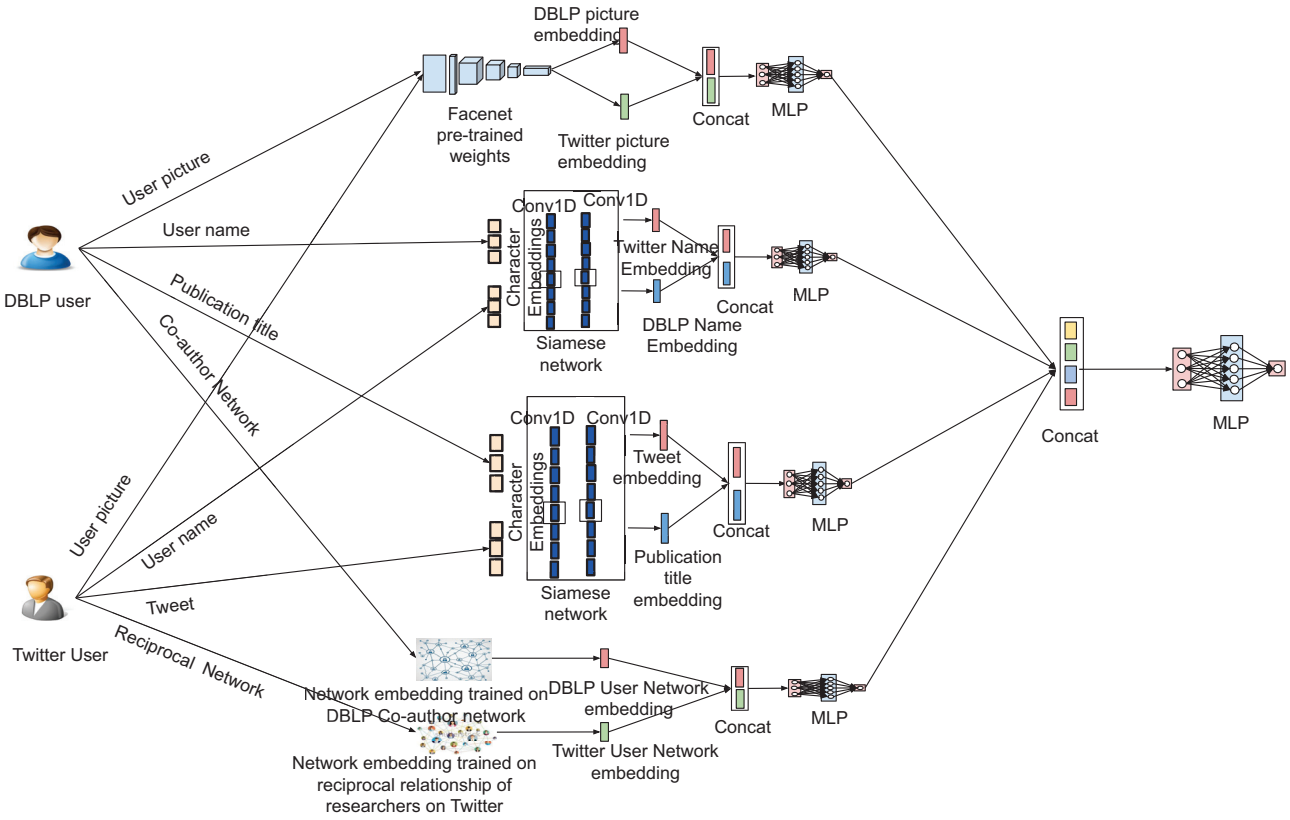

Figure 7: Our end-to-end user identity linking system that jointly models image, text and network modalities.

- a novel data set of name pairs and a Siamese network was constructed to learn and solve user name matching
- novel dataset of (tweet, publication title) pairs was constructed and used with a Siamese network to compute content level matching
- state-of-the-art work (Man et al. 2016) was re-implemented to perform network matching

Results of the unified model are given in the experimental result section.

Experiment

In Eq 1 we defined the user identity linking problem as a binary classification task. In this section, we will will describe our experiment setup and show the results of our experiments. The evaluation metrics used are: precision, recall, F-1 measure and accuracy. All experiments were performed with 10-fold cross-validation. All our experiments were implemented using Keras (Chollet and others 2015) with Tensorflow (Abadi et al. 2016) back-end.

Image level matching: Embeddings for images are generated using the OpenFace (Amos, Ludwiczuk, and Satyanarayanan 2016) implementation which is based on FaceNet (Schroff, Kalenichenko, and Philbin 2015). we used

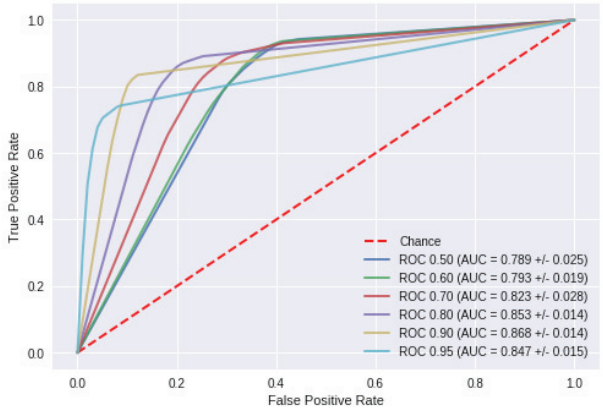

Figure 8: ROC curves with 10 fold cross validation for image level matching with different thresholds.

the publicly available pre-trained weights from OpenFace with the version "nn4.small2.v1.h5" [4]. Our image level matching component takes a pair of images as input. Using pretrained weights from OpenFace, it generates embeddings and learns an MLP classifier with a binary cross en-

[4]https://krasserm.github.io/2018/02/07/deep-face-recognition/

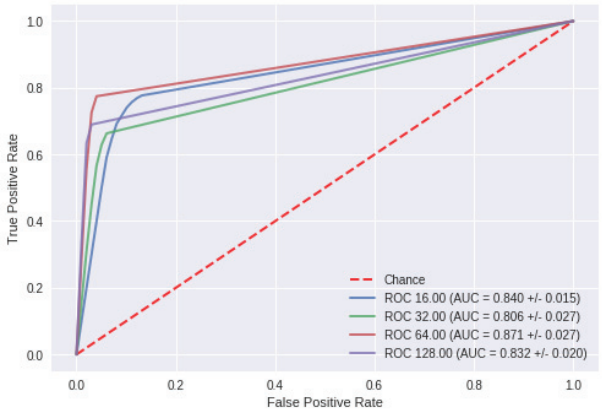

Figure 9: ROC curves with 10 fold cross validation for network level matching with different embedding sizes.

tropy loss on these embeddings to learn whether the input pair is a match or not. This MLP consists of 3 layers: 128 by 64 by 1. The units in the first two layers use Relu activation. We used a dropout of 0.5 for regularization.

Figure 8 shows the performance of image based profile linking for different levels of cut-off values. Image based identity matching performs well, however as we saw in the dataset section not all users have images. At cut-off value of 0.9, it achieves a weighted average: precision 0.93+/-0.01, recall 0.89+/-0.03 and F1-score 0.90+/-0.02.

Network level matching: For our network level matching experiments we use a 3 layer MLP model where each layer uses relu activation with a dropout of 0.25 followed by batch normalization. We used a binary cross entropy loss. The input layer consisted of 64 units and the network was trained using a learning rate of 1e-3, with a batch size of 64 for 25 epochs. Adam optimizer was used.

Node2Vec has several parameters such as the walk length, dimensions of embedding vector, number of walks per node, window size etc. In Figure 9 we see the effect of embedding size on identity matching using network embeddings. With 64 embedding size, using network only. Table 3 presents the network based linking. For network-based matching, authors with overlapping co-authors are hard to distinguish.

Name based matching: Table 2 shows the result of name based matching. Our error analysis for name based prediction shows that the model is generally good but misses cases where names are written in different languages or nicknames and non-obvious abbreviations are used.

Content level matching: The neural network used for matching content embeddings used a Siamese convolutional neural network with an input layer of size 200, learning rate of 1e-3, trained on binary cross entropy loss with Adam optimizer. The network uses a dropout of 0.3 between input and hidden layer and dropout of 0.5 between hidden and output layer. This is trained with a batch size of 64 for 5 epochs.

In Table 4 we see the performance of content based matching. Content based matching suffers from: (i) not every researcher posts tweets about their work (ii) co-authors that tweet or retweet about joint works have matching content and their articles causing false positive matches.

	Precision	Recall	F1-score
matches	0.70 +/- 0.01	0.93 +/- 0.01	0.80 +/- 0.01
non-matches	0.99 +/- 0.00	0.93 +/- 0.00	0.96 +/- 0.00
macro avg	0.84 +/- 0.01	0.93 +/- 0.01	0.88 +/- 0.01
micro avg	0.93 +/- 0.00	0.93 +/- 0.00	0.93 +/- 0.00
weighted avg	0.95 +/- 0.00	0.93 +/- 0.00	0.94 +/- 0.00

Table 2: User identity linking using names.

	Precision	Recall	F1-score
matches	0.82 +/- 0.04	0.75 +/- 0.05	0.78 +/- 0.02
non-matches	0.96 +/- 0.01	0.97 +/- 0.01	0.97 +/- 0.00
macro avg	0.89 +/- 0.02	0.86 +/- 0.02	0.88 +/- 0.01
micro avg	0.94 +/- 0.00	0.94 +/- 0.00	0.94 +/- 0.00
weighted avg	0.94 +/- 0.00	0.94 +/- 0.00	0.94 +/- 0.00

Table 3: User identity linking with network embeddings.

	Precision	Recall	F1-score
matches	0.82 +/- 0.01	0.65 +/- 0.02	0.72 +/- 0.02
non-matches	0.84 +/- 0.01	0.93 +/- 0.01	0.88 +/- 0.01
macro avg	0.83 +/- 0.01	0.79 +/- 0.01	0.80 +/- 0.01
micro avg	0.83 +/- 0.01	0.83 +/- 0.01	0.83 +/- 0.01
weighted avg	0.83 +/- 0.01	0.83 +/- 0.01	0.83 +/- 0.01

Table 4: User identity linking with content.

	Precision	Recall	F1-score
matches	0.94 +/- 0.01	0.86 +/- 0.02	0.90 +/- 0.01
non-matches	0.98 +/- 0.00	0.99 +/- 0.00	0.99 +/- 0.00
macro avg	0.96 +/- 0.01	0.93 +/- 0.01	0.94 +/- 0.01
micro avg	0.97 +/- 0.00	0.97 +/- 0.00	0.97 +/- 0.00
weighted avg	0.97 +/- 0.00	0.97 +/- 0.00	0.97 +/- 0.00

Table 5: User identity linking using profile, network and content.

Table 6 shows the results of state-of-the-art approaches for user-identity matching. Our joint model that puts profile, content and network level features together, brings a significant improvement and achieves the best performance.

Conclusion

In this paper, we have described the user identity matching problem across online social networks. We proposed a joint model that leverages profile, network and content level features which improves the state-of-the-art results that use only some of these features. Our approach is based on current advanced in deep learning that automatically learn feature-representations of images, names, and short texts. One aspect that will be of importance to investigate in future

Method	Precision	Recall	F1-score
Content based (Goga et al. 2013)	0.83 +/- 0.01	0.83 +/- 0.01	0.83 +/- 0.01
Name based (Liu et al. 2013)	0.95 +/- 0.00	0.93 +/- 0.00	0.94 +/- 0.00
Network based (Man et al. 2016)	0.94 +/- 0.00	0.94 +/- 0.00	0.94 +/- 0.00
Our joint model	**0.97 +/- 0.00**	**0.97 +/- 0.00**	**0.97 +/- 0.00**

Table 6: Comparison of our user identity linking system against state-of-the-art results.

work is the effect of the different fields. Most of these fields are dynamic. The bio description of a user changes as a person changes their career. Network information also evolves over time. A good next step is to investigate and take into account these evolving timeline patterns to model the user identity matching task.

References

Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 265–283.

Amos, B.; Ludwiczuk, B.; and Satyanarayanan, M. 2016. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science.

Chollet, F., et al. 2015. Keras. https://keras.io.

De, A.; Bogart, C. M.; and Collins, C. S. 2012. Detecting impersonation on a social network. US Patent 8,225,413.

Fox, J. E., and Jones, D. 2013. Migration, everyday life and the ethnicity bias. *Ethnicities* 13(4):385–400.

Goga, O.; Lei, H.; Parthasarathi, S. H. K.; Friedland, G.; Sommer, R.; and Teixeira, R. 2013. Exploiting innocuous activity for correlating users across sites. In *Proceedings of the 22nd international conference on World Wide Web*, 447–458. ACM.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 855–864. ACM.

Hadgu, A. T., and Jäschke, R. 2014. Identifying and analyzing researchers on twitter. In *Proceedings of the 2014 ACM conference on Web science*, 23–32. ACM.

Hu, B.; Lu, Z.; Li, H.; and Chen, Q. 2014. Convolutional neural network architectures for matching natural language sentences. 2042–2050.

Iofciu, T.; Fankhauser, P.; Abel, F.; and Bischoff, K. 2011. Identifying users across social tagging systems. In *Fifth International AAAI Conference on Weblogs and Social Media*.

Kazemi, V., and Sullivan, J. 2014. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1867–1874.

Kim, J. 2018. Evaluating author name disambiguation for digital libraries: a case of dblp. *Scientometrics* 116(3):1867–1886.

Kong, X.; Zhang, J.; and Yu, P. S. 2013. Inferring anchor links across multiple heterogeneous social networks. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 179–188. ACM.

Labitzke, S.; Taranu, I.; and Hartenstein, H. 2011. What your friends tell others about you: Low cost linkability of social network profiles. In *Proc. 5th International ACM Workshop on Social Network Mining and Analysis, San Diego, CA, USA*, 1065–1070.

Liu, J.; Zhang, F.; Song, X.; Song, Y.-I.; Lin, C.-Y.; and Hon, H.-W. 2013. What's in a name?: an unsupervised approach to link users across communities. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 495–504. ACM.

Liu, L.; Cheung, W. K.; Li, X.; and Liao, L. 2016. Aligning users across social networks using network embedding. In *IJCAI*, 1774–1780.

Lu, Z., and Li, H. 2013. A deep architecture for matching short texts. In *Advances in Neural Information Processing Systems*, 1367–1375.

Mahrt, M.; Weller, K.; and Peters, I. 2014. Twitter in scholarly communication. *Twitter and society* 89:399–410.

Man, T.; Shen, H.; Liu, S.; Jin, X.; and Cheng, X. 2016. Predict anchor links across social networks via an embedding approach. In *IJCAI*, volume 16, 1823–1829.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Perito, D.; Castelluccia, C.; Kaafar, M. A.; and Manils, P. 2011. How unique and traceable are usernames? In *International Symposium on Privacy Enhancing Technologies Symposium*, 1–17. Springer.

Pujari, S. C.; Hadgu, A. T.; Lex, E.; and Jäschke, R. 2015. Social activity versus academic activity: a case study of computer scientists on twitter. In *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business*, 12. ACM.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823.

Sharma, N. K.; Ghosh, S.; Benevenuto, F.; Ganguly, N.; and Gummadi, K. 2012. Inferring who-is-who in the twitter social network. *ACM SIGCOMM Computer Communication Review* 42(4):533–538.

Shu, K.; Wang, S.; Tang, J.; Zafarani, R.; and Liu, H. 2017.

User identity linkage across online social networks: A review. *Acm Sigkdd Explorations Newsletter* 18(2):5–17.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Zagoruyko, S., and Komodakis, N. 2015. Learning to compare image patches via convolutional neural networks. 4353–4361.

Zeiler, M. D., and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*, 818–833. Springer.