Computer Science and Information Systems 20(3):1263–1287 https://doi.org/10.2298/CSIS220110006K

Detecting and Analyzing Fine-Grained User Roles in Social Media[*]

Johannes Kastner and Peter M. Fischer

University of Augsburg
Universitätsstr. 6a, 86159 Augsburg, Germany
{johannes.kastner, peter.m.fischer}@uni-a.de

Abstract. While identifying specific user roles in social media -in particular bots or spammers- has seen significant progress, generic and all-encompassing user role classification remains elusive on the large data sets of today's social media. Yet, such broad classifications enable a deeper understanding of user interactions and pave the way for longitudinal studies, capturing the evolution of users such as the rise of influencers.

Studies of generic roles have been performed predominantly in a small scale, establishing fundamental role definitions, but relying mostly on ad-hoc, data set-dependent rules that need to be carefully hand-tuned.

We build on those studies and provide a largely automated, scalable detection of a wide range of roles. Our approach clusters users hierarchically on salient, complementary features such as their actions, their ability to trigger reactions and their network positions. To associate these clusters with roles, we use supervised classifiers: trained on human experts on completely new media, but transferable on related data sets. Furthermore, we employ the combination of samples in order to improve scalability and allow probabilistic assignments of user roles.

Our evaluation on Twitter indicates that a) stable and reliable detection of a wide range of roles is possible b) the labeling transfers well as long as the fundamental properties don't strongly change between data sets and c) the approaches scale well with little need for human intervention.

Keywords: Social Media, User Role Detection, Classification, Clustering, Supervised Learning Unsupervised Learning.

1. Introduction

As a significant share of personal and public life is shifting to social media platforms, they are growing in terms of user number and activity. The interaction of users can have a profound affect in both social media (eliciting reactions, spreading information) as well as in the real world (driving popular sentiment, affecting political decisions).

While the number of users is huge, their behavior and impact on others are clearly not uniform, thus motivating thorough studies. The need to counter malicious activities has driven many of those studies, providing tools to detect -among others- bots, bullies, spammers and fake news providers in large numbers and with little human intervention.

Yet, these are rather blunt, limited tools that do not provide a deeper understanding of the rich activities and varied user groups present in social media. Studies that do such

[*] This is an extended version of `https://doi.org/10.1007/978-3-030-85082-1_23`

wider and fine-grained investigations on user behavior are indeed performed, but typically require a significant amount of human involvement to organize the data and interpret the results. Thus, they are mostly done in an academic environment on limited sets of users representing coarse-grained roles.

Machine-aided identification of user roles in social media at scale and speed promises interesting insights on their prevalence and impact, allowing to capture different aspects of activity, social media usage, popularity and influence: Not every user that generates large numbers of tweets has malicious purposes but also could only be sharing relevant information to others using his/her network. Forwarding information may be driven by the desire to share relevant information or to endorse certain positions. Not every user that has a large number of followers is a star or influencer, as they may lack in activity. The same social media may be used for information dissemination, but also for conversations or restricted types of feedback.

Automatically recognizing such fine-grained roles provides another benefit, as user classifications can now be performed over longer periods of time. Such a stable recognition provides the means to explain how individual users and communities evolve over time.

We propose a method that combines unsupervised learning to discover fine-grained classes of users over a wide range of features with supervised learning - generalizing expert knowledge from manually labeled reference data to new data sets, mapping role candidates to well-known roles or identifying new roles.

The paper provides the following contributions:

- Our method covers both learning the structure of user groups as well as assigning suitable labels.
- A study on large, complementary data sets shows that both recognizing and transferring roles is feasible over longer time periods or topic variations.
- The classification hierarchy and the cluster metrics support (also iterative) human review, so that identification itself requires little human intervention.
- Sampling strategies provide means to scale the method to large data set as well as provide insights on the certainty and stability of role assignment.

The remainder of this extended paper is structured as follows: In Section 2 we discuss related work. We introduce our methodology in Section 3.2 and provide more details on structure discovery and labeling in Sections 4 and 5, respectively. After an extensive evaluation (Section 6), we conclude the paper.

2. Related Work

Clearly, identifying user roles has been one of the textbook examples of classifier algorithms, yet the application to social networks has been limited to particular aspects. Often, the studies focus on detecting specific roles or describing only a small number of coarse-grained classes. Considering the negative dynamics of many social networks, most researchers focus on identifying specific malicious users, example include: detection of bots [2] or spammers [14], identification of aggressors in the context of cyber bullying [1,11] or –of particular interest recently– discovery of instigators and spreaders of fake news [16,7]. In contrast, our goal is to comprehensively assign all users to roles. Multi-role approaches such as Varol et al. [18], Rocha et. al [6] and Lazaridou et. al [13] limit themselves to identify a small number (often 3-5) of major, coarse-grained groups, roughly corresponding the upper levels of our detection hierarchy. Du et. al [5] provide a somewhat higher number of rules (still lower than ours), but only give generic descriptions. All of these previously mentioned methods are constrained on just detecting the structure by unsupervised learning: clustering via K-Means [13], EM [6] or via topic models [5], leaving the analysis entirely to human experts. In terms of classification, Varol et al. [18] fully rely on such human expertise, using similarity matrices and handcrafted rules. In contrast, qualitative works like Tinati et. al [17] or Java et. al [10] provide a comprehensive overview on fine-grained roles and their semantics, but consider only general rules on how to detect them. An interesting, complementary direction is the work on content communities/web forum, often exploring complex temporal models, e.g., [8]. It should be noted that all of these works (with the exception of [5] (Weibo, 12K users), [11] (Instagram, 18K users), and [8] (Stack Overflow)) solely rely on Twitter due to the limited availability of data from other services. A recent work by Hacker et al. [9] comes closest to our approach, while tackling the -more constrained- problem of user role identification in Enterprise Social Networks. Like our work, it follows a process-based approach involving and aiding human analysts in discovering and interpreting user roles. It applies a wide set of user features and employs clustering to identify user group candidates. As the authors themselves recognize, their problem is less challenging due to the smaller scale and better observability (allowing for more expressive metrics) and more well-defined and less context-dependent roles. Furthermore, we provide a more extensive process by incorporating a classifier to perform knowledge transfer of user role between data sets and employ a sample combination strategy for probabilistic roles assignment and better scalability.

While probabilistic clustering is well-established for centroid methods [4] and recent work presents probabilistic density-based methods with constraints (Lasek et al. [12]), hierarchical clustering is not covered well regarding probabilistic assignment.

3. Research Questions & Approach

Before introducing the main aspects of our approach we want to provide some basic assumptions and definitions:

In the scope of this work, we consider *social media* that allows users to publish content (which we call messages) and organize themselves in structures (networks, groups). These networks enable rich means of interaction on top of both content and structure, such as resharing or conversations. As a consequence, we do not consider media that is purely driven by opaque algorithms such as TikTok.

Users are all types of distinct entities that may visibly interact with the social media, including both humans and algorithms/bots.

A *data set* in our model is a set of messages by users stemming from a single social media, often corresponding to specific events or topics. These messages are recorded and extracted from a social network, currently mostly Twitter for due to its open nature.

As the related work only describes instances of user roles, but not the concept of a role itself, we use the following, basic definition:

A *user role* is a group of users that share similar feature values and are well separated from other groups. The features gather salient properties of users and allow a meaningful categorization, typically capturing behavior and position in the network/media. Groups constitute roles if they are present in sufficient number within a data set and reoccur over multiple data sets.

3.1. Research Questions

Motivated by the introduction in Section 1, we phrase three questions in order to classify diverse user roles in large data sets:

1. To which extent can clusters of users be utilized to sensibly detect user roles in social media and build a classifier to (semi-)automatically label them?
2. Can this approach be applied individually over a wide variety of data sets, currently stemming from the same social media?
3. Can the knowledge on roles be transferred from a (set of) well-understood data set(s) to new data sets?

3.2. Approach

To answer these questions, we introduce our main approach using a high-level overview of our model, which can be seen in Fig. 1.

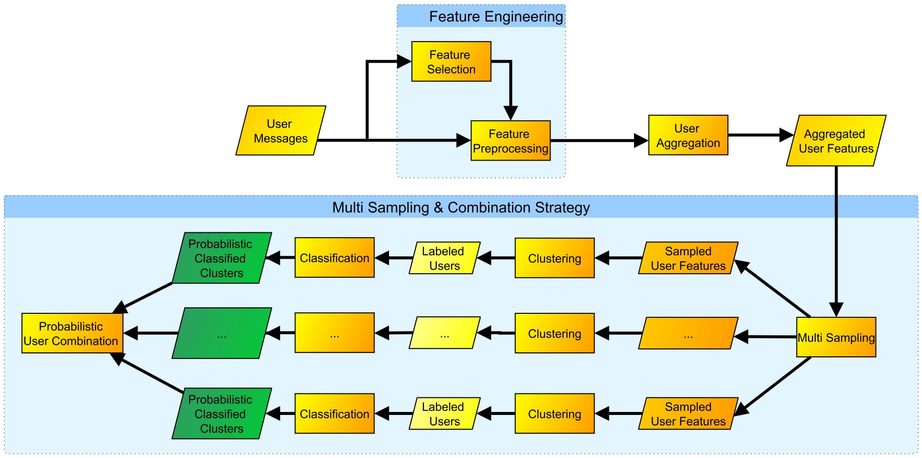


Fig. 1. Flowchart of the approach

Our process starts with a *Raw Data Set*, consisting of messages that were recorded from a social media. In the next step (*Feature Engineering*) we determine the relevant features to capture the various properties of users. As it is our goal to analyze large-scale data sets, it is essential that features are based on widely available data (e.g. not requiring the full social graph) and can be computed at scale, not requiring high complexity and runtime. Clearly, this needs to be repeated for each distinct social media, but - as our experience shows so far - only minor adaption is needed for data from the same social media.

After choosing the features, they need to be *preprocessed* to suit the requirements of clustering and classification methods, including but not necessary limited to outlier removal and normalization/standardization.

In order to solve the competing goals of scalability, minimal human effort as well explainability for fine-grained roles, we devised a multi-sampling strategy that allows us to apply precise and flexible, yet costly clustering methods such as hierarchical agglomerative clustering on large data sets. By gradually expanding the coverage of samples we can turn our analysis from an overall discovery of the general role structure in the data set to a complete assignment of all users to roles. Yet the most important benefit is enabling hard, hierarchical clustering and classifications methods to produce probabilistic assignment, capturing the uncertainties of role allocation for users on the fringes between groups.

Therefore, instead of clustering and classifying a data set once (which can be very costly and does not capture uncertainties well), we create representative samples with controllable overlap with our *Multi Sampling* strategy. These samples are clustered hierarchically, creating candidates for user roles that can be explained from the features in the clustering tree. With this *Multi Sampling* strategy we are able to enrich the hierarchical agglomerative clustering with aspects of probabilities, while commonly available method only allow for hard assignment.

The cluster analysis is followed by the *classification*, which delivers for each sample clusters of users with probabilities to given user roles from literature.

The competing labels from the different clusters and classifiers are *combined* to produce a *probabilistic role assignment*, so that we are able to clearly recognize the core users of clusters (same role assignment) as well as users which lay in between different clusters and thus user roles (different role assignment). The fact that some users do not get covered by the Multi Sampling strategy or occur only once is a tuneable, which is explained more in detail as part of our analysis in Section 6.

Since we have addressed different use cases in our questions, we have to distinguish between complementary scenarios, requiring different quantities of human involvement given the amount reference data: completely/partly unexplored data sets without or little training data vs well-established training data. This distinction is emphasized in the program flow chart in Fig. 2 that serves as a guidance through the following sections. While steps, such as the preprocessing of the raw data set, which includes normalization and standardization techniques, as well as the sampling and clustering of the data remain identical for both scenarios, the differences are as follows.

1) If only data sets such as a new social network or not yet comprehensive training data are available, we discover groups of similar users and their hierarchical relationship by clustering, thus providing candidates for user roles. The analyst will then assigning role labels to these groups to build manually new training data or enrich already available training data. He/she is aided by quality metrics, visualizations and dimensionality reduction like Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) to inspect the assigned labels. In turn, these manually provided labels form the input for a classifier that captures this knowledge and can be cross-validated on this data set.

2) If a sufficiently complete training data from the same social network with the same features is available for a classifier, this -possibly very tedious- labeling process can be cut short by providing candidate labels for the clusters in a new data set. Our training set (and additional manual labels) may be cross-validated to ensure the quality of the model.

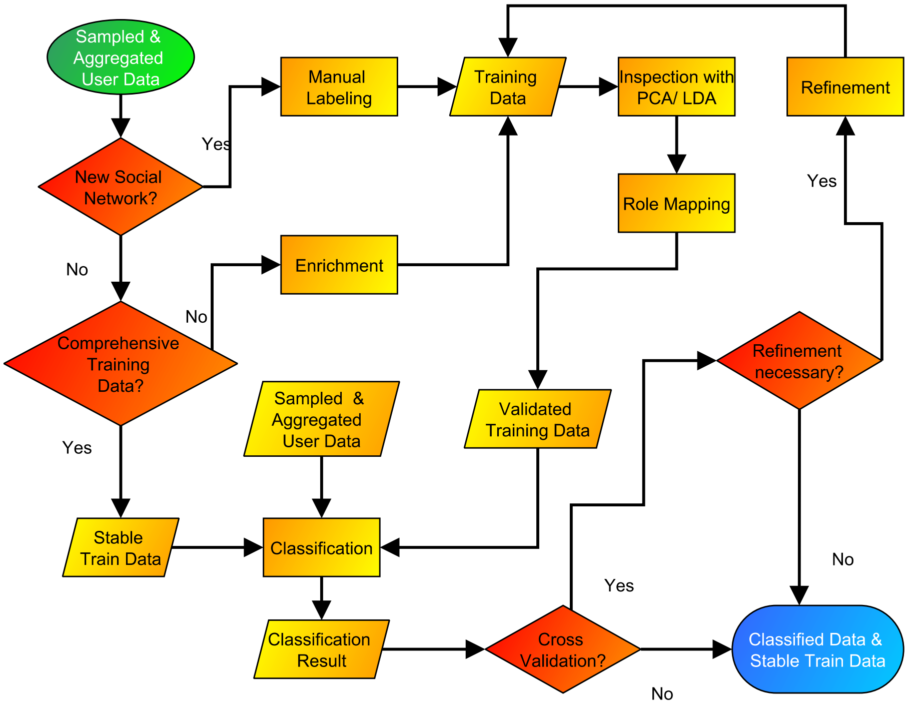


Fig. 2. Flowchart of the classification considering the scenarios

The user can evaluate these candidates either within the new data set or compare the roles across the data sets, as we show in our analysis. We also explored causes of mislabelings and provide methods to adapt them, yet a full exploration of options remains future work.

For each scenario we can go on with the combination strategy of the clustered and classified users and analyze the results considering different tunables in the steps of the multi sampling and combination strategy as well as the classification step which will be presented in detail in Section 6.

4. Feature Selection and Data Clustering

After introducing the main aspects and questioning of our approach, we focus now on the steps of the Feature Engineering and Data Clustering

4.1. Feature Engineering

In this work we aim to use features that cover significant and complementary aspects of users and are well established in the literature [6,13,1]. In addition, it should be feasible to compute in large scale so that data is commonly available and incur moderate cost to compute. Likewise, we want to avoid a large number of features, as this hurts both algorithm performance and explainability.

Fig. 3 highlights the classes and instances of features: *static user properties* express (self-)description: most relevant is the *verified* status of a user, traditionally reserved for celebrities and influential users. *User activity* is characterized by the number of original tweets of each user (observed and “offtopic”), the activities on other tweets such as retweets and replies within the topic as well as mentions of other users. Basic *network position* features like the number of *followers* and *followees* of a user as well underpin the potential to exert influence. In turn, the user’s ability to actually elicit *reactions from the network* is captured by the *ratio of tweets* to lead to *replies* and *retweets* as well as the frequency of *being mentioned*.

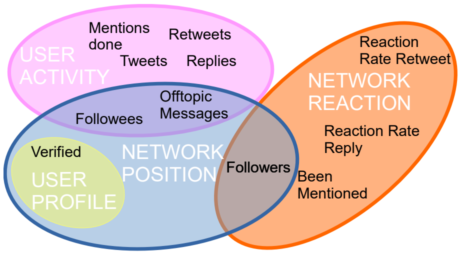


Fig. 3. User Feature Classification

We investigated a wide variety of additional features from these classes, but dropped them as they were correlated or had little discriminative power. We excluded complex network metrics such as centralities, spatio-temporal features [18] as well as content analyses [1,11]. Those suffer from data availability as well as cost and may be used for refinement or specialized sub-roles. Even partial social graphs are exceedingly hard to get from any social media (including Twitter), while our crawling strategy already provides a topic focus.

To investigate the correlation of the features described in the last paragraph, we depicted the correlation of pairwise features in a symmetric heat-map, as can be seen in Fig. 4. The bar on the right hand-side visualizes if pairwise features have a high negative correlation (deep blue), no correlation (white) or a high positive correlation (deep red). As most of the feature pairs have no correlation or only a weak positive or negative correlation, the features *followers* and *followees* are the most correlated features (as popular users show gains in either dimensions), yet changes in their ratio turned out to be a discriminative feature for specific groups, so we still considered both. Likewise, we keep some feature pairs with moderate positive correlation, e.g. *mentions done* / *tweets*, *offtopic messages* / *retweets* as well as *followers* / *offtopic messages*.

Given that many features in social media exhibit significant skew and value domain variation, we normalize each data set individually, so that the relative distribution differences and feature drifts are captured. More specifically, we reduced skewness using logarithmic transformation, followed by a Min-Max normalization to bring the values into

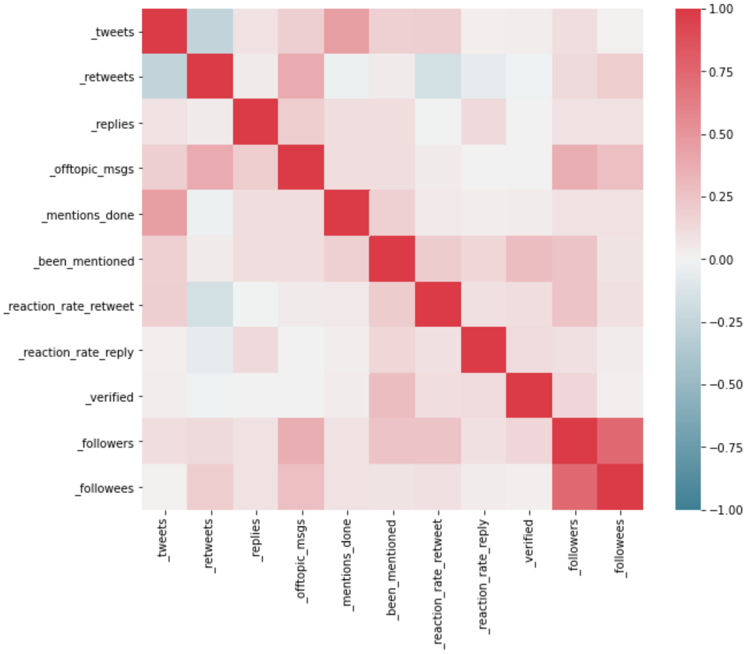


Fig. 4. Correlation Matrix for User Features

a range of 0 to 1. We also considered methods like inverse transformation, square/cube root and box-cox, but neither resulted in more balanced results.

Table 1 shows the properties of the *Olympics 2012* data set features before and after the normalization and standardization process.

As can be seen in Table 1 most of the features excluding *offtopic messages*, *reaction rate* for *retweets* and *replies* and the *verified* status contain strongly right skewed data, which can also be seen in the small median values up to to the high 99th percentile and maximum. The normalization strategy is effective, leading to almost balanced skewness and median values.

4.2. User Group Clustering

To identify the structure and (sub)-groups among the user data, we evaluated a broad range of unsupervised learning approaches based on centroids (e.g., K-Means[1]), density (like DBScan[2]) and probability distribution (e.g., EM[3]. Hierarchical clustering[4] turned out to be most suitable: a) it can capture complex, irregular shapes without requiring a fixed number of clusters and b) the hierarchy serves as an (yet unlabeled) classification tree on

[1] https://scikit-learn.org/stable/modules/clustering.html#k-means

[2] https://scikit-learn.org/stable/modules/clustering.html#dbscan

[3] https://scikit-learn.org/stable/modules/mixture.html

[4] https://docs.scipy.org/doc/scipy/reference/cluster.hierarchy.html#module-scipy.cluster.hierarchy

Table 1. Feature Statistics Olympics 2012 Twitter Data

Feature	**Original Features**				**Normalized**		
	Median	**99%**	**Max**	**Skew**	**Median**	**99%**	**Skew**
tweets	2	19.00	16621	543.97	0.11	0.31	0.72
retweets	1	13.00	3780	331.82	0.08	0.32	0.75
replies	0	3.00	759	204.19	0	0.21	2.52
offtopic	59	3790	>100K	13.28	0.35	0.71	-0.19
ment._done	0	8.00	7802	723.10	0	0.25	2.26
been_ment.	0	3.00	>140K	1017.85	0	0.12	6.73
react._retweet	0	0.75	1	2.55	0	0.75	2.55
react._reply	0	0.33	1	8.67	0	0.33	8.67
verified	0	0.00	1	22.62	0	0	22.62
followers	172	9520	>15M	229.46	0.31	0.55	-0.07
followees	231	2908	>600K	74.67	0.41	0.6	-0.97

which feature differences explain the user roles. Since we have a multi-dimensional data set geometric linkage methods, in particular Ward's worked best.

Yet, hierarchical methods are not without issues: On a technical side, they tend to incur high CPU and memory costs even for moderately large data sets due to their $O(n^2)$ scaling. On a conceptual side, almost all popular approaches tend to only support "hard" clustering, assigning a data point exclusively to a group. In reality, users may exhibit trait of multiple roles to various degrees, making a "soft", probabilistic assignment more meaningful.

To address both of these issues, we chose a sampling/ensemble-based approach when clustering the data: Clustering a small number of samples allows us to quickly discover the structure while drastically reducing the cost compared to clustering the whole data set. By incrementally drawing more samples, we see a linear cost increase (while allowing parallel execution) and provide a faithful representation of the data. With overlapping cluster results from several samples for the same user, we can choose to assign to a majority role or the probability for specific roles. Likewise, we can determine how stable the role recognition is. The number of samples becomes a tuneable, trading off the effort of computation (and labeling) with the coverage of users and the amount of support for the roles. If all users need to be covered, we may minimize the overlap or apply metric-based assignments.

4.3. Determining Cluster Count & Analysis

A key question when identifying user groups (and thus roles) by clustering is the actual number of such groups. While hierarchical clustering avoids the issue of having to provide a fixed number of clusters beforehand (as, e.g., K-Means or EM require), the classification tree (often represented as a dendrogram) produced by the clustering allows for a large range of cluster numbers - between 1 and the number of input values, as the cluster candidates are aggregated along the hierarchy.

Traditionally, this issue is tackled by computing quality metrics such as Davies-Bouldin, Silhouette and Calinski Harabasz (which are all internal cluster quality measures) for cluster candidates determining a useful point in this metric space, e.g., at diminishing returns using the elbow method. We followed the approach of [19] which relies on the distances of the dendrogram as metric and refines the elbow with the acceleration of these global and local distances.

This approach yielded already useful, but not entirely satisfactory results: we could reliably determine the generalized, coarse-grained main groups, which correspond well to those in Fig. 5. For fine-grained roles we (often) did not get clear indications or (sometimes) groups that would not relate to user roles described in the literature.

We augmented this generic approach with a domain-specific methodology that is based on the insight that user roles often can be refined by clear differences on specific features, not just on general, global metrics. Intuitively, comparing boxplots (which show means, median and quantiles) in the manual labeling process led us to determine features whose differences explain the characteristics of subgroups. To formally express and discover these differences, we rely on statistical measures. In particular, we utilized effect sizes such as (pooled) Cohens d [15] to capture significant feature deviations. Cohens d is defined as the difference between the means of two sets divided through the standard deviation, while the pooled standard deviation [3] allows to deal with cluster candidates of different sizes, so smaller clusters with significant features can be detected reliably. Otherwise such smaller clusters that represent pronounced user roles such as *Star* or *Semi star* tend to get absorbed by bigger clusters. Furthermore, pooling is less sensitive to feature drift.

The refinement process is modeled using a Depth-First search covering the subtrees in the dendrogram forming the generalized roles. At a search step, the process compares (in pairwise fashion) the measures for each feature of the current cluster to those of its two direct descendants which are the refinement candidates. This search continues as long as there are significant effects, leading to a possible cutoff for refinement in this particular path. After the whole Depth-First Search we only have to cut off at the deepest distance in the dendrogram where we have found a clustering with considerable features.

The significance criterion remains a tuneable, but in most cases the results were best when finding at least 2 features with a large effect. When we investigate new data sets the criteria for overall significance may have to be adjusted, yet -on our experience- this rarely needed, as in almost all cases these values delivered useful clusterings.

Considering how clustering are used in the overall approach, no perfect fit for the cluster number is actually necessary. Instead we would like to slightly overestimate the number of clusters, avoiding an early cutoff that would lose possible user groups. The spurious groups will be merged either during the manual label assignment or by the trained classifier, as shown in the following section.

5. User Role Identification

While the hierarchical cluster structure identifies candidates, it does not provide the actual user roles. We now describe the (manual) assignment that also serves as the training data for a role classifier as well as the transfer of this user role knowledge to new data sets. This Section provides more details on the scenarios which were introduced in Section 3.2 as well as in Fig. 2.

5.1. (Manual) Role Assignment

Considering that neither a general consensus on types of user roles in social media nor precise definitions or models exist (see Section 2), we apply several complementary methods to derive meaningful candidates. Starting from the cluster hierarchy (described in Section 4) that provides indications on the (approximate) number of clusters and their respective separation, but no meaning of those, we apply complementary approaches: 1) manually analyzing the overall structure of the clustering (dendrogram) and features of the individual cluster (boxplots) with the significant features allows us to match these clusters to fine-grained user roles from the literature [17,10], e.g., *Semi Star* or *Amplifier* (cp. Fig. 8). 2) dimensionality reduction such as PCA or LDA (cp. Fig. 7) aids this exploration process in several ways: the composition of the main components further highlights relevant features. The reduced number of dimension aids the computation cluster separation metrics and simplifies visual inspection. Likewise, it also helps with correlating user roles across data sets and exposes the drift/evolution.

These approaches yield an iterative strategy: Using the stopping heuristics, the number of clusters is narrowed to typically 15-30 candidates, though this value is clearly dependent on the specific data set. The structure of the dendrogram guides the manual mapping process in which we compare the feature distributions presented in boxplots. Certain heuristics support this work: 1) Specific classes of roles tend to manifest themselves further up the hierarchy, creating subtrees for those classes that could then be refined into more specific roles. 2) Some very distinct roles tend to show up in most data set, providing an "anchor" for the labeling. The refinement process is stopped once we do not gain additional, well-discernible or well-interpretable clusters. In some cases it may be useful to coarsen the roles again or combine several clusters into a single role.

We match these aspects to the role descriptions provided when possible (in particular on well-studied roles like *Star* with its large number of followers, almost always verified status and generally high impact despite relatively low activity), but also observed stable, recurring clusters that did not align well with the known roles descriptions, leading to role discovery. In our data set, we also found more *action triggering* user roles (cp. Fig. 5) such as *Idea Starters*, which are similar to *Semi Stars*, but gain popularity in the network by creating more content and triggering higher reactions in the network. Furthermore there are *Amplifiers*, which are well networked users pushing and spreading mostly (existing) trends and *Rising Stars*, which gain a large number of followers by activity in the network, receiving significant reactions in terms of retweets, but not yet at the level of *Stars* or *Semi-stars*, which fit into the intermediate user role group.

More intermediate user roles are *Spammers*, which have mostly a high activity in the network but are not as popular as the action triggering users. They are similar to *Average Users*, which can hardly be distinguished from the whole data set focusing on deviations of the statistical indicators of the features and stand in most cases one of the biggest groups in the user roles. *Daily Chatters* distinguish from the spammers because of their more moderate action in the network. In most cases they lay in between the *Spammers* and *Average Users*. Furthermore there is a role of the *Commentator*, which is similar to the *Daily Chatter*, but is more active in creating content, retweeting and especially in cases of reactions in the network by replying to content.

The last group of similar users we recognized in the data sets are passive users like *Forwarders*, who are better networked like average users, but mostly only forward content and thus receive only less reactions in the network. *Listener*, who mostly only consume, and thus have a weak connection in the network, share only less content and do not trigger other users. They are only underbid by *Loner*, who are mostly inactive in the network.

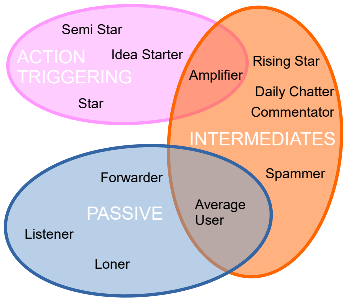


Fig. 5. User Roles

The same process can be applied across data sets, comparing the dendrograms, cluster metrics/descriptions and correlating the labeled roles. By doing so, we can track the user roles across the data sets and evaluate concept shifts and drifts among them, such as their frequency/probability of roles or their feature distributions. Typically, we observed around 10-15 class candidates that did show up in varying frequency over our data sets, sometime disappearing entirely.

While this manual labeling excels at capturing the specific knowledge of a domain expert and produces high-quality and well-described role clusters, it is a tedious process with very limited scalability that may suffer from reproducibility issues due to the subjective nature of human assessment.

5.2. Classification

The goal of classification is to transfer knowledge on user roles from existing data sets or other samples of the same data sets that have already be labeled. The multi-sampling approach as well as the previous clustering stage lead to some particularities that we describe in more detail.

The classifier consumes two types of input: On the one hand the training data, described in the previous subsection, is essential to capture the user role models. On the other hand the clustered user data, which needs to be classified so that each cluster is assigned to a user role label. As mentioned before in 3.2 we designed a Multi Sampling and Combination Strategy to provide scalability and role probabilities for each user a stable user roles based on the identification of significant features. In Fig. 6 the important parts of our Multi Sampling and Combination Strategy can be seen.

We start with user data that has been aggregated and engineered into representative features. The full set is split up in several representative, possibly overlapping samples that are clustered and form the second input for the classification. When applying manual labeling (as outlined in the section before), each user receives a role label depending on the cluster it belongs due. For those clusters that are not manually labeled, the classification process provides for each cluster, and thus for each user in the cluster, a probability vector to the given user roles. After each user in each sample has been clustered and classified, we can combine all probability vectors for each user into a single probability vector. While

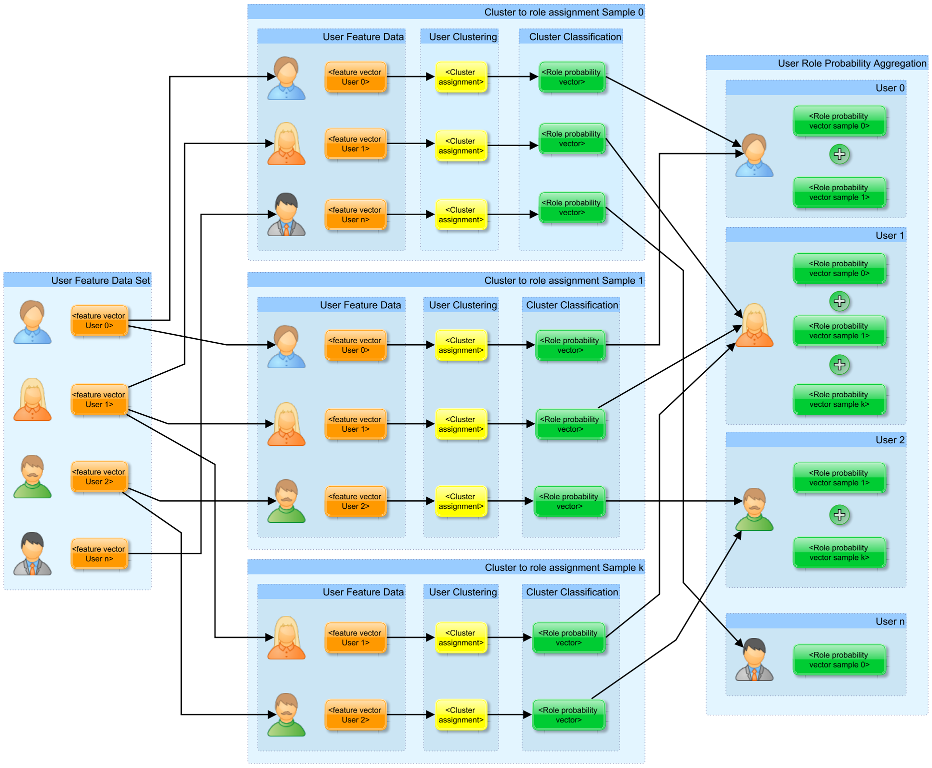


Fig. 6. Combination Strategy using classifier with training data available

it is feasible to only retain the user role with the highest probability at either the individual classification results or after the combination, this leads to suboptimal results, as the uncertainties for users that are at the fringe of two or more clusters and thus did not get a clear majority for one user role. Beyond that, the variances samples may lead to different clusters (and thus relative user behavior), further strengthening the need for a probabilistic classification to ensure accuracy for borderline users.

To overcome the issues mentioned at the end of Section 5.1, we utilized a classifier that was trained from several samples of one data set using the cluster means and determined the role labels on clusters in other data sets, expressing a n-class problem. For training, we took samples that showed the best cluster separation (supported by PCA, which can be seen in Fig. 7) to minimize the noise in the model and concatenated them. As initial experiments showed, the original number of dimensions in the data yielded better quality than reduced dimensionality.

The creation of training data was a time consuming iterative process consisting of a manual cluster analysis followed by a manual classification of each cluster to the user roles from literature. We picked several cluster centroids from several samples for each user role and adjusted the training data incrementally after using PCA and LDA. If we have a closer look at the training data in Fig. 7 the reduced dimensions redeem the complexity of our given multidimensional cluster means. For each user role we have clearly separated training data as the dimension reduced projection in Fig. 7 shows for almost all roles. Since we have some user roles which lie close in between two user roles, e.g., *AVG User* vs. *Forwarder* vs. *Daily Chatter*, the process of creating training data is a very important key element which demands a manually consequent and precise procedure.

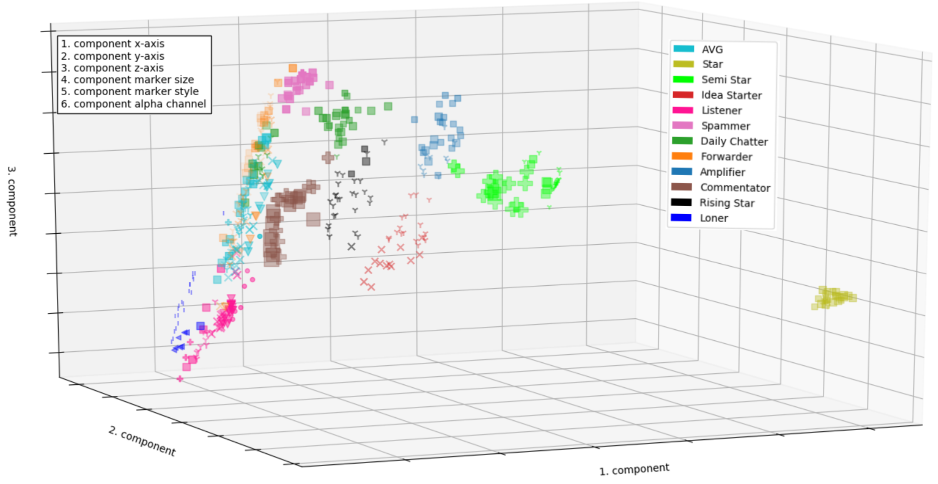


Fig. 7. Principal component analysis of clustered samples from *Olympics 2012* data set

We also investigated the tradeoff between classifying individual users instead of entire clusters. While the former may yield higher classification performance (no clustering is needed when working with a new data set), we observed that the inherent “noise” shown by individual users does not lend itself well to either training or classification. We therefore decided to represent a cluster by feature values computed from all its members. For this step, we considered the arithmetic mean as well as the median, each also with pooled Cohens d (relative to the entire sample) to boost separation. Our evaluation showed that means tended to provide better separation than median, while pooled Cohens d seemed to capture more temporal evolution than “pure” means.

As our (clustered) data sets are relatively small and skewed, yet we seek to express a large number of classes, we see little support for some classes. This more or less rules out deep learning. Instead, methods based on ensembles of decision trees, e.g., Gradient Boosted Decision Trees (GBM) or Extremely Randomized Trees (ET), multi-class support-vector machines (SVM) or k-nearest-neighbor (kNN) turned out to be most suitable. We utilized the Python implementations of scikit-learn for ET, SVM and kNN as well as XGBoost[5] for GBM.

The setup to build training sets utilized repeated stratified cross validation with three splits (leave-one out, due to the small amount of data) and three repetitions (with different permutations to cater for possibly missing groups). We used F1-macro as a metric to

[5] https://xgboost.readthedocs.io/en/latest/

compensate for class imbalance and prevent focus on either precision or recall and applied grid search to tune parameters. All classifiers learn and generalize well, leading to 94-95 percent score in validation and training set with no obviously stronger or weaker candidates.

When transferring the classification to new data sets, we compensated for mislabelings by varying training and prediction data (e.g. cluster number) or choosing more suitable training sets. Explicitly including drift models and relevance feedback from the user remain future work.

6. Evaluation

After explaining the idea, concepts and set-up of our approach in Section 3 we now present and discuss results of experiments on diverse data sets from Twitter. In our analysis we address the three questions outlined in Section 3.2. For each step we do not only show the technical results but also report our empirical observations.

6.1. Data Sets and Preparation

While our long-term goal is to recognize user roles over a variety of data from various social media, we focused in this initial analysis on data sets that are well-defined and contain a large number of users. As in most of the related work, we relied solely on Twitter, as its one for the few social media services in which data sets containing large numbers or longer periods of time are available.

In order to transfer knowledge on user role detection, we are looking at several classes (Table 2): major sports tend to be repetitive and predictable with a very large number of messages and users, covering significant periods of time. Different types of sports provide (albeit limited) thematic variance. These data sets are complemented by those of two major disasters which also tend to have a strong, yet very different topic focus and different interaction patterns. Finally, we applied our work on an instance of the Twitter sample stream to assess a data set without a strong topic focus.

Table 2. Overview on Data Sets

Data Set	Messages	Users	Time Period	Category
Olympic Games 2012	13.68M	2.27M	August 2012	sport event
Olympic Games 2014	14.58M	1.96M	February 2014	sport event
Olympic Games 2016	38.05M	4.76M	July/August 2016	sport event
FIFA World Cup 2014	109.00M	10.40M	June/July 2014	sport event
2015 Paris Attacks	6.77M	0.74M	November 2015	tragic incidence
NFL Superbowl LIV 2020	8.89M	0.89M	2. March 2020	sport event
2016 Berlin Truck Attack	0.66M	0.15M	19. December 2016	tragic incidence

Our data sets had each been recorded using the Twitter Streams API and Search API using commonly proposed hashtags. We only considered users that were active at least twice, as several metrics require aggregations. Generally speaking, the relative feature distributions after normalization varied only slightly over time from 2012 until today, with minor changes: users tend to move slightly more into “reactive” behavior of forwarding than content generating or mentioning, while the *verified* status is now much more prevalent. Overall activity increased moderately, forwarding actions became more widespread.

6.2. Initial Data Set: 2012 Olympics

The first step focuses on a single data set (*Olympic Games 2012*) with uniform feature usage and role stability due to the relatively short period of time. Given those benign conditions, these analyses provide insight to which extent such as clustering, user roles detection and automated labeling are feasible, as stated in Q1 in Section 3.1.

Following the approach outlined in section 3.2, we created samples covering 5% to 10% of the data set and applied the hierarchical clustering we introduced in Section 4.2 afterwards using `scipy.cluster`. The latter sample size represents the maximum that could be clustered on the machines available on an 8-core partition of an AMD Epyc 7401. A small data set like *Berlin 2016* may still be clustered completely, yet a sample can be generated almost instantly, as can be seen in Table 3. For large data sets, full clustering is clearly impossible, while samples fit well. The cost is almost entirely consumed by creating the linkage matrix, so refinement/exploration steps are interactive in all variants. After clustering, we manually labeled clusterings of the samples to get a ground truth as training and test data as mentioned in Section 5.1. In real-life settings, this labeling and testing may be performed incrementally until a sufficiently good understanding of the data has been established.

Table 3. Runtime and memory of samples, full data sets and approximated(*)

	Oly12 5%	**Oly12 10%**	**Oly12 100%**	**Berlin16 10%**	**Berlin16 100%**
runtime	19 min	136 min	226 h*	10s	38 min
memory	94 GB	375 GB	375 TB*	1.2GB	184 GB

In particular after applying PCA (see Fig. 7), we can identify a number of well-separated clusters. Despite showing some minor variances, the dendrograms (see Fig. 8) over the set of samples exhibit a very similar overall structure that has become a part of our overall classification as on the leftmost column of Table 4: there are between 3 and 5 subtrees representing major groups, expressed by very distinctive feature values: The first major group (green) shows users that are able to *trigger strong reactions* (*retweets*, *replies*, *being mentioned*), the second (red) shows *passive users* with fairly weak positions in the network, while the group(s) in between show various degree of *moderate activity and impact*. Even further down the tree, (as shown on the boxplots), we see a strong motivation for fine-grained roles. While the cluster sizes are often small, there are salient feature differences (which we can detect using statistical tests like Cohens d) that explain the existence and semantics of this group. In the example one can see how *Semi Stars* and *Amplifiers* split on (among others) *retweet* activities and *reactions*. Overall, we determined 12 roles in the Olympics 2012 data set that are described in Table 4. Some characteristics are shown in the second column, in particular stronger deviations from the average as well

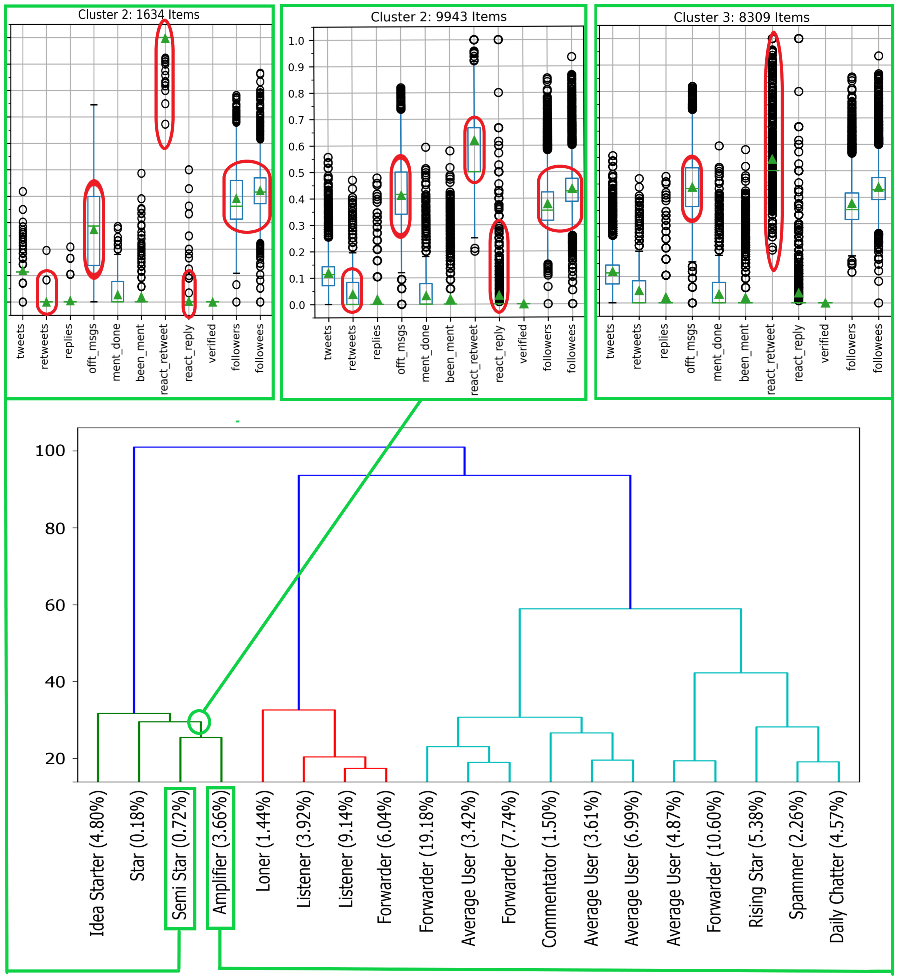


Fig. 8. *Olympics 2012* 10% sample Dendrogram with salient features

as (broadly) similar user groups. Note that these user roles are not strictly defined for all data sets, but a good starting point for the evaluation of all further data sets.

Continuing in our pipeline from Fig. 1 with the combination of the clustered and classified data sets and thus the users (see also Fig. 6), we now take a closer look at the coverage and certainty of roles, which can be seen in the subfigures of Fig. 9 for combinations of two different samples sizes (5% vs. 10%) for the *Olympic 2012* data set and the *Superbowl 2020* data set (10% vs. 20%).

When increasing the number of samples and thus the number of individually classified users the number of role assignments per user improves. As a result, the number of users without any role assignment (red bar) drops quickly, while the number of users with multiple, mostly consistent role assignment (green bar) grows rapidly. Furthermore, the number of users that only see a single role assignment (orange bar) becomes smaller,

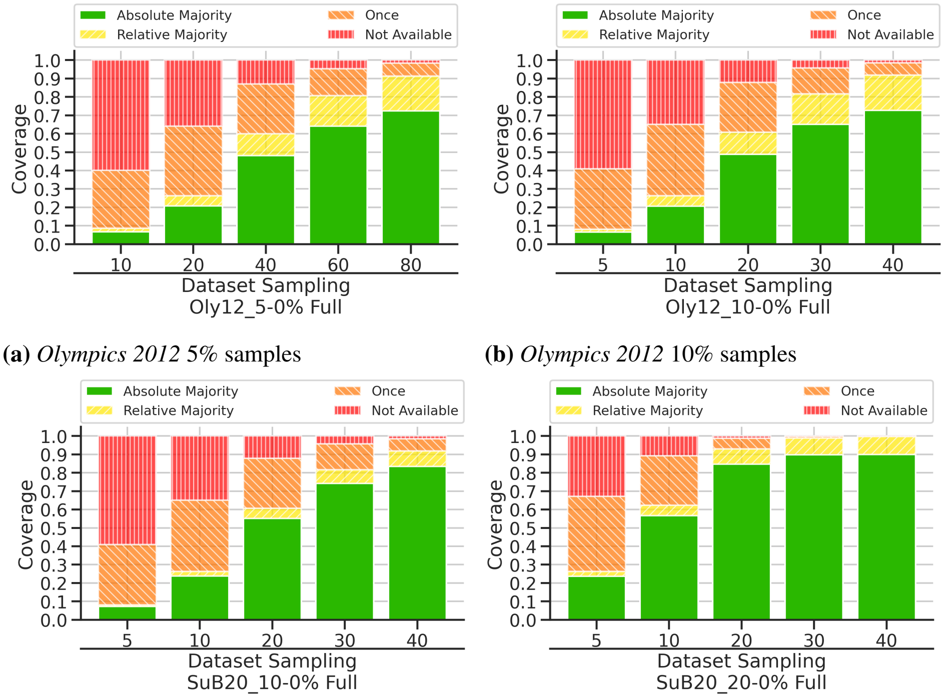
Table 4. User roles and their characterization: ≈ shows closeness to other roles, ↓/↑ feature deviation from close role/whole data set, ↘ / ↔ / ↗ changes over time

	Role	Characteristics	Freq./Trend
action triggering	Star	followers > followees, verified, ↓ activity, ↑ mentioned	0.2–0.8 ↗
	Semi Star	≈ Stars, ↓ followers, mentioned, ↑ react. (re)tweet, retweets, replies	0.2–1.4 ↘
	Idea Starter	≈Semi Star, ↓followers, ↑reactions	1–4 ↔
	Amplifier	≈ Idea Starters, Semi Stars, ↑ followers, followees	0.5–5 ↘
intermediates	Rising Star	≈ Semi Star, Idea Starter, Amplifier ↑ followers, (re)tweets, replies	1.5–5.5 ↘
	Daily Chatter	≈ Average User, Spammer, ↓(re)tweets, offtopic	5–15 ↔
	Commentator	↑ replies, offtopic, reations	0.3–2 ↘
	Spammer	↑ (re)tweets, replies, offtopic ↓ followers, followees, reactions	1–7 ↔
passive	Average User	offtopic > tweets, retweets	8–30 ↓
	Forwarder	retweets > tweets, ↑ offtopic, followers, followees. ↓ reactions	25–65 ↑
	Listener	↓ (re)tweets, reactions	6–20 ↗
	Loner	↓↓ tweets, offtopic, followers	0–1.5 ↘

enabling us to perform actual probabilistic assessments on the assignment certainty. In turn, the increasing "relative majority" part (yellow bar) gives insights on user that are not well identified - which is data set-dependent, but often includes *Spammer*, *Loners*, etc.. For most of these roles the percentage for the strongest role is between 40 and 50%, which is also a very persuasive value in the context of a 12-class classification problem. Also the distance to the second-best user role is quite high, which also substantiates the significance of user roles in our sampling and combination strategy. Further increasing the number of samples does not significantly decrease the share of those users, indicating that theses are not artifacts of the sampling approach. Overall, the scaling works well, thus validating our approach.

When utilizing bigger, yet fewer samples (Fig. 9b and 9d) compared to the combination of smaller samples (Fig. 9a and Fig. 9c) for the same number of users, the quality of the results tends to be slightly better (in particular for the *Superbowl 2020* data set), yet at much higher resource requirements due to the quadratic complexity for clustering. As a consequence, it is typically better to utilize smaller, but more numerous samples.

We evaluated the clustered and labeled samples (in total 507 clusters) with the classifiers mentioned in Section 5 and achieved nearly perfect results with the classifiers, as the leftmost data points in Fig. 11b show. The confusion matrix between roles (Fig. 12a - 12d) confirms these results, as the are only very few mislabelings resp. misclassifications between *Average User*, *Daily Chatter* and *Listener*, respectively - which are also close to each other in feature space. The strong variance in the feature distribution present in the boxplots (Fig. 8) also shows why training and classifying individual users instead of clusters yields inferior results. Since we use a sampling and combination strategy, the



(a) *Olympics 2012* 5% samples

(b) *Olympics 2012* 10% samples

(c) *Superbowl 2020* 10% samples

(d) *Superbowl 2020* 20% samples

Fig. 9. Sampling and combination for several datasets

effect of mislabeling or misclassification is dampened by the fact that the first role is -in most cases- very dominant or -in cases of no majority- has a significant distance to the second-best role.

Overall, the results show that both clustering and classification work well. Expert knowledge is needed to interpret the dendrogram and assign roles, but already within a single data set, the knowledge can be transferred to additional samples and their clusters.

6.3. Multiple Individual Datasets

The second step analyzes several data sets individually to understand if the approach is more widely applicable (thus providing insights on Q2). Furthermore, this will show us of the same or similar roles are present in data sets varying in time and topic and how they evolve over time.

The 12 user roles identified on the *Olympics 2012* data set are also present and well-separated in the other data sets, though -as the rightmost column of Table 4 shows- the frequency (in percent) varies over data sets (and over time):

In the *Olympics 2014* (278 clusters) and *FIFA World Cup 2014* (193 clusters) data sets very few changes can be observed: *Average User* and especially *Loner* occur less frequently, while *Forwarder* and *Listener* occur more frequently.

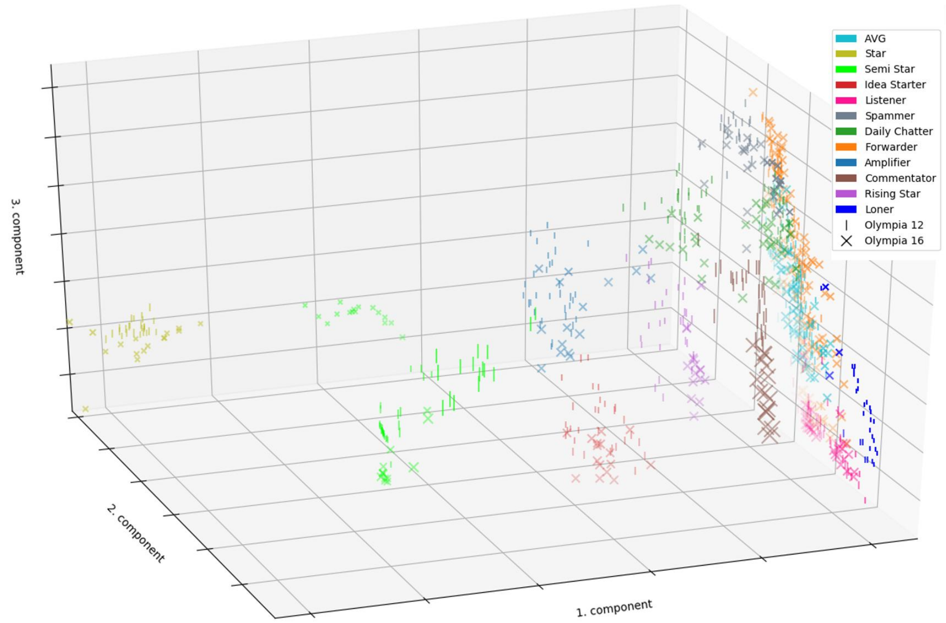


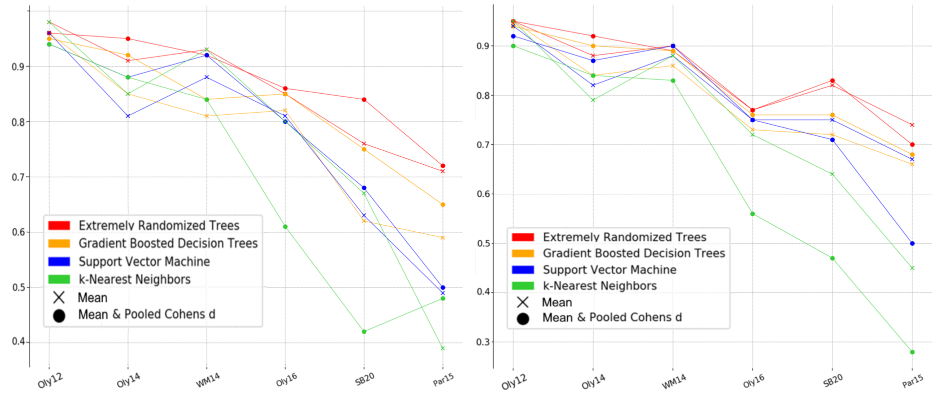
Fig. 10. PCA of clustered samples from *Olympics 2012* vs. *2016*

The first significant changes in terms of user roles frequency and features occurred in the *Olympics 2016* data set (355 clusters). The PCA in Fig. 10 -showing cluster centroids of our training data- provides a salient concept drift for many user roles between the *Olympic 2012* data set (Pipe symbol |) and the *Olympic 2016* data set (Crosses X): in particular, *Semi Stars* tend to also cover a space much closer to *Stars*, as the *"verified"* status was more freely distributed by Twitter. The already observed trend on the frequency of *Average Users* /*Loner* and *Forwarders* strengthens, as many users prefer to retweet more than creating their own content. This trend continues for the *Superbowl 2020* (345 clusters) data set, which is otherwise (despite the different sports and the time difference) rather similar to *Olympics 2016*.

The *2015 Paris Attacks* (160 clusters) data set covers a very different topic and distinct interactions (fewer *offtopic messages*, more *retweets*). Some user roles are not present (*Commentator*, *Loner*), yet most of the overall trends match the picture of the "sports events": forwarding instead of content creation becomes more dominant (both as feature and as role), corresponding to the wider trend in all social media. In turn, "influencer" roles become pronounced, to the point where the *Semi Star* may have to split into two separate sub-roles.

The only exception where we could not apply our methodology was the random *Sample Stream*, as features based on topics lose their usefulness.

For Q2, we could show that the same features can be applied, leading to consistently recognizable user roles. We could observe how the distributions of roles shift over time and also correlate the roles of users over the boundaries of the data sets. Yet, at this step, the effort of labeling samples of each data set manually is a limiting factor.



(a) Macro Average F1. **(b)** Weighted Average F1.

Fig. 11. Information retrieval measure F1 for classifiers

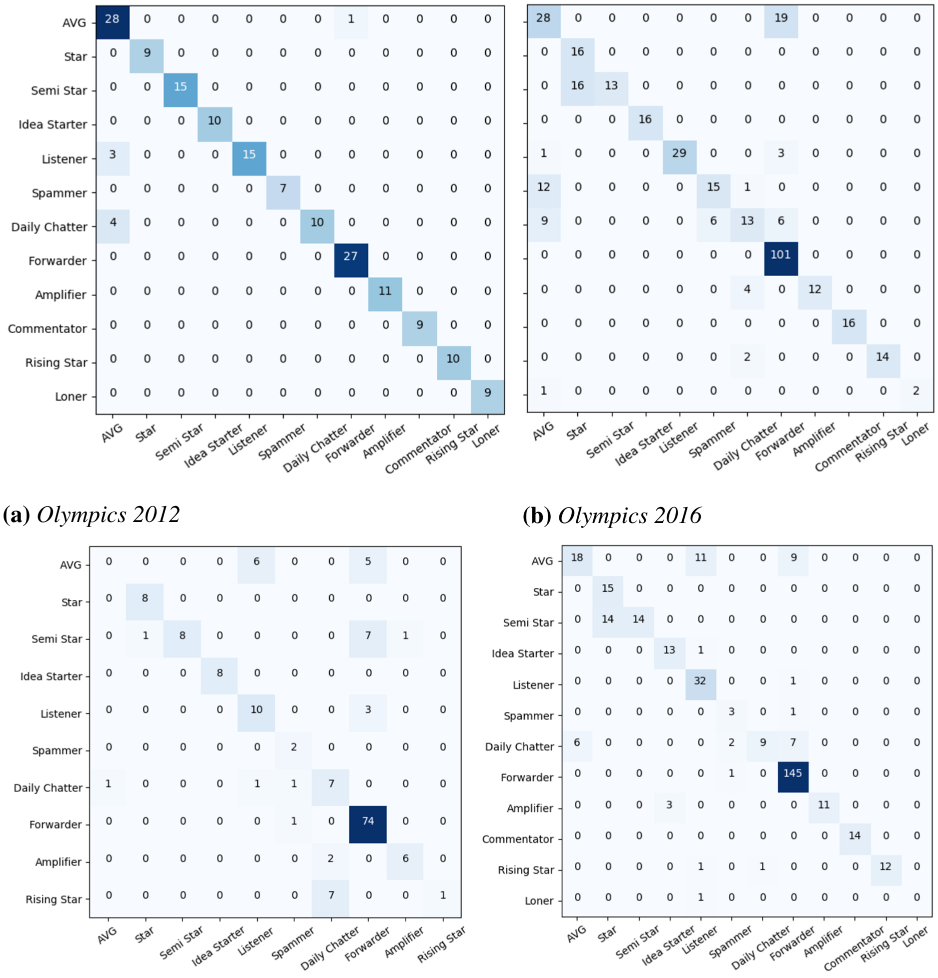
6.4. Applying Models on New Data Sets

In the third step, we classify new data sets with the models gathered from reference data to transfer role knowledge and assess the quality and effort involved, answering Q3. We further study the impact of variation and drift to understand the limitations.

Fig. 11a and 11b show the F1 scores when classifying the data set based on the *Olympics 2012* as the reference, as it provides the longest prediction period. While the weighted values in Fig. 11b depicts the quality of frequently represented user roles, the macro values in Fig. 11a support the overall performance of the classifiers. Overall, one can see a gradual degradation over time on the sport events, as the classification methods do not explicitly capture the drifts observed in the previous section, but still generalize the roles over time. Yet, the best methods achieve a 0.85 F1 score for "late" sport events. The *2015 Paris Attacks* data set sees the largest degradation, showing topic and interaction differences have a more profound impact than time. When comparing all these results to the slightly worse "macro" values, one can see that small groups are captured well, while larger clusters tend to be somewhat "blurry".

kNN and SVC keep up well for short time intervals, but tend to lose ground on longer distances. ET holds a small edge over GBM, while the latter stays still competitive and incurs much lower runtime cost. Both benefit from enriching the data sets with the pooled Cohens d values.

The confusion matrices for the *Olympics 2012*(Fig. 12a) *Olympics 2016*(Fig. 12b) and *Superbowl 2020* (Fig. 12d) data sets show how that roles that were either not well separated in the *Olympics 2012* data or drifted significantly are most affected. Yet, these misclassification often leads to adjacent roles, e.g., *Average Users* as *Listener* and *Forwarder*, *Daily Chatter* as *Forwarder* and *Average User* or *Semi Stars* as *Stars*. Especially the scores for Superbowl 2020 emphasize the drift to forwarding content as well as the rise of influencers from *Semi Stars* to *Stars*. Only in the *Paris 2015* data set (Fig. 12a) some more misclassifications are noticeable, due to the topical different training data. Thus the



	AVG	Star	Semi Star	Idea Starter	Listener	Spammer	Daily Chatter	Forwarder	Amplifier	Commentator	Rising Star	Loner
AVG	28	0	0	0	0	0	0	1	0	0	0	0
Star	0	9	0	0	0	0	0	0	0	0	0	0
Semi Star	0	0	15	0	0	0	0	0	0	0	0	0
Idea Starter	0	0	0	10	0	0	0	0	0	0	0	0
Listener	3	0	0	0	15	0	0	0	0	0	0	0
Spammer	0	0	0	0	0	7	0	0	0	0	0	0
Daily Chatter	4	0	0	0	0	0	10	0	0	0	0	0
Forwarder	0	0	0	0	0	0	0	27	0	0	0	0
Amplifier	0	0	0	0	0	0	0	0	11	0	0	0
Commentator	0	0	0	0	0	0	0	0	0	9	0	0
Rising Star	0	0	0	0	0	0	0	0	0	0	10	0
Loner	0	0	0	0	0	0	0	0	0	0	0	9

(a) *Olympics 2012*

AVG	Star	Semi Star	Idea Starter	Listener	Spammer	Daily Chatter	Forwarder	Amplifier	Commentator	Rising Star	Loner
28	0	0	0	0	0	0	19	0	0	0	0
0	16	0	0	0	0	0	0	0	0	0	0
0	16	13	0	0	0	0	0	0	0	0	0
0	0	0	16	0	0	0	0	0	0	0	0
1	0	0	0	29	0	0	3	0	0	0	0
12	0	0	0	0	15	1	0	0	0	0	0
9	0	0	0	0	6	13	6	0	0	0	0
0	0	0	0	0	0	0	101	0	0	0	0
0	0	0	0	0	0	4	0	12	0	0	0
0	0	0	0	0	0	0	0	0	16	0	0
0	0	0	0	0	0	2	0	0	0	14	0
1	0	0	0	0	0	0	0	0	0	0	2

(b) *Olympics 2016*

	AVG	Star	Semi Star	Idea Starter	Listener	Spammer	Daily Chatter	Forwarder	Amplifier	Rising Star
AVG	0	0	0	0	6	0	0	5	0	0
Star	0	8	0	0	0	0	0	0	0	0
Semi Star	0	1	8	0	0	0	0	7	1	0
Idea Starter	0	0	0	8	0	0	0	0	0	0
Listener	0	0	0	0	10	0	0	3	0	0
Spammer	0	0	0	0	0	2	0	0	0	0
Daily Chatter	1	0	0	0	1	1	7	0	0	0
Forwarder	0	0	0	0	0	1	0	74	0	0
Amplifier	0	0	0	0	0	0	2	0	6	0
Rising Star	0	0	0	0	0	0	7	0	0	1

	AVG	Star	Semi Star	Idea Starter	Listener	Spammer	Daily Chatter	Forwarder	Amplifier	Commentator	Rising Star	Loner
AVG	18	0	0	0	11	0	0	9	0	0	0	0
Star	0	15	0	0	0	0	0	0	0	0	0	0
Semi Star	0	14	14	0	0	0	0	0	0	0	0	0
Idea Starter	0	0	0	13	1	0	0	0	0	0	0	0
Listener	0	0	0	0	32	0	0	1	0	0	0	0
Spammer	0	0	0	0	0	3	0	1	0	0	0	0
Daily Chatter	6	0	0	0	0	2	9	7	0	0	0	0
Forwarder	0	0	0	0	0	1	0	145	0	0	0	0
Amplifier	0	0	0	3	0	0	0	0	11	0	0	0
Commentator	0	0	0	0	0	0	0	0	0	14	0	0
Rising Star	0	0	0	0	1	0	1	0	0	0	12	0
Loner	0	0	0	0	1	0	0	0	0	0	0	0

(c) *Paris 2015* **(d)** *Superbowl 2020*

Fig. 12. Confusion matrices of classifications

F1-scores actually understate the quality of the result, as they do not take the adjacency of roles into account.

We added the data set of the *2016 Berlin Truck Attack* (Christmas market) that was not evaluated in the previous stages and provides topic similarity to *2015 Paris Attacks*, while being close to the *Olympics 2016* in time. This data set provides a good opportunity to assess the impact of different training sets: in addition to baseline of the *Olympics 2012* and close sets (*Olympics 2016*, *2015 Paris Attacks*) and *Superbowl 2020* as a small, recent data set, we tested two combinations: *Olympics 2012* and *Superbowl 2020* for the full time range and *2015 Paris Attacks* with those two as mix of time range and topic proximity. As Table 5 shows, these combined data sets provide the best results, matching manual classification or producing misclassifications to close roles. *2015 Paris Attacks* by itself seems to be too small to provide a sufficiently general model, but is able to boost the full time range model.

The experiments show that a transfer of labeling knowledge is effective with certain limitations: large topic differences or very long time differentials diminish the usefulness, yet a good choice of reference data can mitigate this effect.

Table 5. Classification of several data sets

Classifier	Oly12	Oly16	Par15	SB 20	Oly12 + SB20	Oly12 + SB20 + Par15
XGB	0.58	0.59	0.51	0.70	0.78	0.92
ET	0.74	0.63	0.56	0.73	0.77	0.82

7. Conclusion & Future Work

In this paper we proposed a method on how to determine and label user roles in large-scale social media data sets. This method combines unsupervised learning (more specifically, hierarchical clustering) to discover the classes of users over a wide range of features and supervised learning - generalizing the knowledge from manually labeled smaller data sets.

Our analysis on a range of large data sets from Twitter show that well-separated roles can consistently be recognized and transferred. The labeling achieves high accuracy not only within the same data set, but also on new data sets from different event types and/or years apart. The resource requirements of such analyses are modest, bringing them in range of commodity hardware.

For future work, we see a number of interesting directions: As the quality of classification begins to deteriorate over longer time frames, we plan to incorporate evolution into both clustering and classification, considering both temporal models (for long-term studies of snapshots) and stream clustering (for short-term, continuous analyses). They may also pave the way for longitudinal studies of users groups and user mobility among groups. Likewise, adapting our model to cope with topically non-related or even topically unconstrained data sets poses a new set of challenges. Initial experiments show that the method should generally work, but significant work still needs to be done. In either case, testing our method on a wider range of data sets from Twitter or even other social networks would be highly interesting.

References

1. Chatzakou, D., Kourtellis, N., Blackburn, J., Cristofaro, E.D., Stringhini, G., Vakali, A.: Mean Birds: Detecting Aggression and Bullying on Twitter. WebSci (2017)
2. Chu, Z., Gianvecchio, S., Wang, H., Jajodia, S.: Who is Tweeting on Twitter: Human, Bot, or Cyborg? In: ACSAC (2010) (2010)
3. Cohen, J.: Statistical Power Analysis for the Behavioral Sciences. Erlbaum, Hillsdale, NJ [u.a.], 2. ed. edn. (1988), literaturverz. S. 553 - 558
4. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39(1), 1–38 (1977), http://www.jstor.org/stable/2984875
5. Du, F., Liu, Y., Liu, X., Sun, J., Jiang, Y.: User Role Analysis in Online Social Networks Based on Dirichlet Process Mixture Models. In: 2016 International Conference on Advanced Cloud and Big Data (CBD). pp. 172–177 (2016)
6. Edgar, R., Alexandre, P.F., Caladoa, P., Sofia-Pinto, H.: User Profiling on Twitter. Semantic Web Journal (2011)
7. Espinosa, M., Centeno, R., Rodrigo, A.: Analyzing User Profiles for Detection of Fake News Spreaders on Twitter - Notebook for PAN at CLEF 2020 (09 2020)
8. Fu, C.: Tracking User-role Evolution via Topic Modeling in Community Question Answering. Information Processing & Management 56(6), 102075 (2019)
9. Hacker, J., Riemer, K.: Identification of User Roles in Enterprise Social Networks: Method Development and Application. Business & Information Systems Engineering 63 (08 2021)
10. Java, A., Song, X., Finin, T., Tseng, B.: Why We Twitter: Understanding Microblogging Usage and Communities. In: WebKDD/SNA-KDD (2007) (2007)
11. Kao, H.T., Yan, S., Huang, D., Bartley, N., Hosseinmardi, H., Ferrara, E.: Understanding Cyberbullying on Instagram and Ask.Fm via Social Role Detection. In: WWW '19 Companion (2019)
12. Lasek, P., Gryz, J.: Density-based Clustering with Constraints. Computer Science and Information Systems 16, 7–7 (01 2019)
13. Lazaridou, E., Ntalla, A., Novak, J.: Behavioural Role Analysis for Multi-faceted Communication Campaigns in Twitter. In: WebSci (2016) (2016)
14. Li, H., et al.: Bimodal Distribution and Co-Bursting in Review Spam Detection. In: WWW (2017) (2017)
15. Sawilowsky, S.: New Effect Size Rules of Thumb. Journal of Modern Applied Statistical Methods 8, 597–599 (11 2009)
16. Shu, K., Zhou, X., Wang, S., Zafarani, R., Liu, H.: The Role of User Profiles for Fake News Detection. In: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. p. 436–439. ASONAM '19, Association for Computing Machinery, New York, NY, USA (2019)
17. Tinati, R., Carr, L., Hall, W., Bentwood, J.: Identifying Communicator Roles in Twitter. WWW '12, rel MSND Workshop (2012)
18. Varol, O., Ferrara, E., Ogan, C.L., Menczer, F., Flammini, A.: Evolution of Online User Behavior during a Social Upheaval. WebSci (2014)
19. Zambelli, A.: A Data-Driven Approach to Estimating the Number of Clusters in Hierarchical Clustering. F1000Research 5 (08 2016)

Johannes Kastner is a researcher and PhD Student at the University of Augsburg (Germany). He received his Masters Degree in Computer Science from the University of Augsburg (Germany) in 2016. His research interests include Clustering, Classification and User Role Detection in general.

Peter M. Fischer holds the chair for Databases and Information Systems at the University of Augsburg (Germany) as a Full Professor since 2017. He received his Diploma/Masters Degree in Computer Science from the Technical University (TU) Munich (Germany) in 2002. He worked as a researcher at TU Munich, University of Heidelberg (Germany) and ETH Zürich (Switzerland), where he recieved his PhD in 2006. After his work as a Postdoc and Senior Researcher at ETH Zürich he became an Assistant Professor for Web Science at the University of Freiburg (Germany). His research interests include the analysis of social streams and graphs, scalable database systems for temporal data, adaption and recommendation of information as well as provenance and assurance of data and operations.

Received: January 10, 2022; Accepted: August 05, 2022.