Building Topic Models to Predict Author Attributes from Twitter Messages

Notebook for PAN at CLEF 2015

Caitlin McCollister[1], Shu Huang[2], and Bo Luo[1]

[1] Information and Telecommunication Technology Center, University of Kansas, USA
[2] Microsoft Research
caitlin.mccollister@gmail.com, shuhuang.psu@gmail.com, bluo@ku.edu

Abstract We use the topic modeling software package MALLET [10] to construct models of 100 topics each for the four languages in the scope of the PAN'15 Author Profiling task. The topics in these models are essentially groups of words that may be semantically related and are frequently observed near each other in a collection of training documents. To ensure we had a sufficiently large body of examples to build such models, we collected our own corpora of Twitter messages in English, Spanish, Italian and Dutch. We also use MALLET to infer the most likely distribution over the generated topics that could have produced any given tweet instance, allowing us to represent tweets as concise 100-element document-topic distribution vectors. These representations serve as inputs to a set of classifiers that make predictions for unknown authors' age, gender, extroversion, stability, agreeableness, conscientiousness, and openness.

1 Introduction

Over time, it is common for a single Twitter user to publish tweets related to multiple aspects of his or her life which may be quite independent of each other. For example, a user might write about his or her professional occupation while at work or attending a conference, post pictures of family members while at home or on vacation, and link to news articles about international politics while reading on the train during the daily commute. Even when examining fewer than one hundred tweets per author, as is the case with the PAN'15 training corpus, most authors' Twitter streams are effectively a mixture of distinct subjects or topics. Our approach to the PAN'15 Author Profiling task [13] is motivated by the expectation that authors will produce language that points to a variety of different or even contradictory traits, and the observation that certain common themes do appear repeatedly even across authors and target classes.

2 Background

2.1 Previous PAN Author Profiling Approaches

Some of the more successful entries in previous years, especially PAN'14 [12], are those that acknowledge the substantial diversity of authors within the target classes for

predictions. The PAN'14 solution by López-Monroy et al [9], one of the top-ranked entries for accuracy in both the English and Spanish Twitter subcorpora, extracts weighted word frequency features from documents and compares the values to those typical of various subprofiles. Those subprofiles are subsets of authors within a target class, such as females aged 18 to 24, who were grouped together by a clustering algorithm based on the most distinguishing words in their writing. By generating and using more fine-grained target classes, the software can train a model that recognizes and accommodates a variety of writing styles and subjects that map to the same original target class.

Another PAN'14 entry by Weren et al [19], which was further refined in a follow-up paper [18], demonstrated the potential effectiveness of information retrieval based features, such as the cosine similarity of a given test document and the labeled training documents. In this implementation, a set of similarity features was found to be more discriminative for age and gender than several common readability measures or the prevalence of dictionary words and punctuation marks. Treating incoming test documents as queries in a document retrieval system and using a combination of aggregate functions on the top-ranked results allows the classification to be based on the most closely related training documents even if many dissimilar documents exist within the correct target class.

2.2 Previous Work in Topic Modeling

Several research groups have pursued the use of topic modeling, including Latent Dirichlet Allocation (LDA) [4], either to gain insight about the processes involved in social media communication [1], or to make predictions about authors and the text they produce online [11]. Schwartz et al have conducted large-scale studies based on millions of English-language Facebook status updates written by tens of thousands of users, and have published several resulting linguistic resources that they claim to be accurate predictors of authors' age, gender, and personality [16]. They were able to collect the status messages, which are often of a similar length to Twitter messages, from volunteers who specified their age and gender directly, and completed a personality profiling questionnaire yielding numeric values for the same "big five" personality traits that we aim to predict in the Author Profiling task. The group has distilled and made available two types of resources based on this work. The first type includes weighted lexica of one- to three-word phrases that are most discriminative for high or low values of the measured personality traits, as well as for several age bins and males versus females. We have made use of these lexica to compute twelve of the features in our "secondary" feature set, described in section 3.1. The other resource made available by Schwartz et al is a set of word clusters consisting of the top 20 words representing topics in an LDA-derived topic model of 2000 topics.

To explore the viability of using topic models to generate features for the kind of data in the Author Profiling task, we initially implemented a set of 2000 features corresponding to the topic word clusters published by Schwartz et al. In these features, we summed the number of words in the cluster that appeared in the tweet at least once, then weighted that sum according to the global term frequencies of the matching words and the length of the tweet. Our initial experiments using these features for English tweet classification looked promising, but faced one significant challenge: the studies

from which the topics and word clusters were derived came from exclusively English-language text, and the features were not particularly useful for our non-English sub-corpora. Conducting additional massive studies on Spanish, Italian and Dutch speaking Twitter users with known gender, age and personality was beyond the scope of our entry in the Author Profiling task. This dilemma inspired the collection of our own unlabeled Twitter corpora for all four languages, with fewer total documents than Schwartz et al used, but more than the number supplied as labeled data in the PAN'15 training corpus. The resulting four corpora are described further in section 3.1.

3 Software Design and Implementation

One of our earliest design decisions was whether to treat all of a given author's tweets as a single body of text, cluster them together by content or in fixed-size chunks, or process them as independent documents all associated with the same author. Our intuition was that the best way to account for high intra-author variation in tweet subject matter and style would be for our software to treat individual Twitter messages as instances in a classification problem, and pool the predictions for all of an author's tweets to make a single prediction per (author, attribute) pair at the end of the testing phase. We implemented and tuned our software for the individual tweet representation, but included a configuration flag to allow concatenating all tweets per author so that we could test the viability of that representation after later software components were completed.

Although treating each tweet as an individual document entails a greater number of predictions to be made in the classification framework, we avoid a potential explosion in dimensionality by limiting the number of features in our models. In the interest of achieving what we felt were reasonable running times within the provided testing environment, especially if the hidden test datasets turned out to be larger than the training datasets, we decided against using the common bag-of-words or n-gram based representations, in which the size of the vocabulary (and thus the feature set) increases rapidly with the number of instances. Instead, we chose to pursue a topic modeling approach in which tweets are encoded as vectors that describe them as an inferred distribution over a fixed-size set of topics generated using the MALLET topic modeling software.

The number of topics in the LDA-based topic model has to be specified at the start of the model training process, so we made our choice of 100 topics after trying both larger and smaller numbers and noting the effect on training time, peak RAM usage, and discriminative power in terms of the computed information gain of the resulting feature sets. While MALLET can supply default values for most of the possible parameters to its particular implementation of LDA, we modified some to suit our application: we set the alpha parameter to 0.5 due to the short document lengths, used 10,000 sampling iterations, and enabled automatic hyperparameter optimization every 50 iterations. These choices were guided by general background literature on topic modeling [17], other studies using MALLET for social media text [16], and eventually by conducting multiple trials using subsets of the training data. Since our topic models are independent of the labeled training datasets provided for the shared task, they only need to be trained once to generate a set of reusable, serialized model files. Even so, we found we could

complete this process on the virtual machine provided to us in the TIRA evaluation framework [6] in under two hours per language.

3.1 Feature Extraction

Primary Feature Set Our topic models are built from datasets of unlabeled Twitter messages which we have collected specifically for this purpose, so that none of the labeled PAN training data is used to define the topics themselves. This was accomplished using the freely-available Twitter corpus-building tool, TWORPUS [2], which can be downloaded and run locally as a web-based application. The application connects to a centralized archive of Twitter message IDs, the user IDs that wrote them, and language tags assigned by a language detection algorithm. Because only the relevant IDs and language tags are stored in the central archive and distributed to TWORPUS users, who then use an included Twitter crawling utility to download the actual message content, the application is compliant with the terms of Twitter's developer agreement forbidding the redistribution of full tweet text and metadata.

We collected four Twitter corpora (one for English, Spanish, Italian and Dutch) spanning the time period from April 2014 to May 2015, with tweets as evenly distributed as possible throughout that period; this was still subject to the availability of the requested number of tweets for each language in the central TWORPUS archive. After retrieving the full text of over 60,000 tweets per language, we used a custom script to remove duplicate or near-duplicate tweets such as simple retweets and bulk-generated advertisements, still leaving over 50,000 tweets per language. No specific action was taken to allow or disallow multiple tweets from any given author; we found that roughly 90 percent of the collected tweets are the only messages collected from their respective authors.

The tweet text from our downloaded TWORPUS corpora needed to be preprocessed in the same fashion that our training and test data would be: we convert all text to lowercase and use the tokenizer included in the CMU Twitter Part-of-Speech Tagger tools [11]. We performed several additional steps on just the model-training input text: we removed lists of language-specific stopwords provided in NLTK [3], and use the Python library Gensim [14] to filter out extremely common or rare terms from our downloaded tweets. In our initial trials of our topic models as classification features, we found that removing such terms from the model-training input resulted in more coherent and discriminative topics.

In the training phase of our software, we again use MALLET to infer the distribution over topics for the labeled training documents that were supplied in the PAN'15 corpus. This yields a 100-element vector for each single-tweet instance. Those topic vectors are used as inputs to train a classifier for each of the 26 (language, attribute) pairs being predicted for the Author Profiling task. In the testing phase, we compute the topic distribution vectors of incoming test documents using the same topic model definitions as we did in the training phase.

Secondary Feature Set In order to establish a reference for how well our topic model features performed on the task compared to more conventional methods, we implemented another set of features in the Python programming language which we could

evaluate alongside our primary set. We built separate models using the two sets of features, used the same preprocessed data as input, and used the same types of classifiers for nominal and numeric target classes. While some of these features are based on published word lists or clusters derived from exclusively English datasets, the presence of emoticons, hashtags and conveniently universal profanities makes most of them still useful even on the non-English PAN'15 subcorpora. Our secondary feature set is described below:

- **Token count and length**. 3 numeric features: Number of tokens, average number of characters per token, maximum number of characters per token in tokenized tweet text.
- **Special word classes**. 4 numeric features: Proportion of words (tokens) containing at least one non-alphabetic character, proportion of words that are URLs, username mentions, or hashtags.
- **Position-specific special word classes**. 6 binary features: Whether the first or last word is a URL, username mention, or hashtag.
- **Special character classes**. 3 numeric features: Proportion of non-whitespace characters that are punctuation, accented alphabetic characters, or digits 0 through 9.
- **Personality and Gender phrases**. 12 numeric features: From the study of Facebook status updates by Schwartz et al [16], we combine the 100 most correlated words, phrases and emoticons for high and low values of the five personality traits being predicted, so that 10 features represent the number of such words present in a single tweet and normalized for tweet length. Similar features were created for typically male or female language elements.
- **VADER Sentiment Analysis scores**. 4 numeric features: Computed using the VADER sentiment analysis library [8]. "Positive" and "Negative" sentiment scores range from 0 to 1, and estimate the proportion and intensity of positive and negative words and phrases. "Neutral" indicates the proportion of sentiment-neutral words in the text. "Compound" is a sum of positive and negative scores, normalized to the range [-1, 1].

The VADER engine is fast, accounts for varying degrees of sentiment polarity, and is designed to handle the informal, short messages of social media text. However, because it makes heavy use of English modifiers and negation structures that are context-sensitive, we only use the sentiment analysis features with the English-language subcorpus.

3.2 Classification and Prediction

Given the above schemes for feature extraction on the training and test datasets, we use the computed feature vectors as inputs to a classifier created for each (language, attribute) pair by calling the WEKA software package [7]. This design choice was motivated by the desire for a framework in which we could experiment with a wide variety of methods for classification and regression without making significant modifications to the data processing and file formatting components of our software. Throughout the

development period, we were able to observe the effects of other optimizations or design choices, such as those in the feature extraction components, when combined with different types of classification and regression models.

We modeled the gender and age group attributes in the PAN'15 Author Profiling task as discrete classification, or what WEKA calls "nominal class" problems. While we did try using various discretization methods to transform the five personality attributes from real-valued regression to classification problems, we did not find a clear advantage to either approach over all four languages and all five attributes. Thus, for the sake of simplicity in implementation, we treated all personality attributes as "numeric class" problems in WEKA.

For both the nominal and numeric classes, we build one of WEKA's "attribute selection" filters into the classification or regression model at the time it is trained, so that the same subset of "attributes" (which we call "features" in section 3.1) will be used on the training and test data. Our motivations for applying a feature selection method at this stage of the software are mostly performance-related. It dramatically decreases the time required for training and testing, and keeps the peak RAM utilization safely within the 4 gigabytes allotted to our virtual machine in the TIRA evaluation framework [6], even if the software is to be evaluated on larger datasets in the future.

Our final configuration choices for the classification and regression components of the software are as follows:

- **Nominal attributes** (age group and gender): FilteredClassifier
  - **Filter**: AttributeSelection using CfsSubsetEval with BestFirst forward search
  - **Ensemble method**: RotationForest [15] using base classifier REPTree
- **Numeric attributes** (personality traits): FilteredClassifier
  - **Filter**: AttributeSelection using CfsSubsetEval with GreedyStepwise forward search
  - **Ensemble method**: Bagging [5] using base classifier REPTree

In the testing (prediction) phase of our software, for each (language, attribute) pair, all feature vectors computed for the testing instances are submitted to the trained WEKA model at once, along with the author ID so that the predictions returned by WEKA can be grouped by author. For example, when making predictions for (English, extroverted), if one of the English authors has 100 tweets in the dataset, the predictions from the WEKA classifier will include 100 floating-point predictions of the author's "extroverted" attribute, ranging from [-0.5, 0.5] as per the Author Profiling task specifications. For numeric attributes (the five personality traits) we take the median value of all the individual predictions. For nominal attributes (age group and gender) we take the discrete class label that occurred most frequently in the individual predictions. This process is conducted once with the WEKA models trained on our primary feature set (document-topic vectors) and again using those trained on the secondary feature set.

The final step in our classification and prediction procedure is to resolve any differences in the predicted values generated by models using the two feature sets. For numeric attributes, we simply take the mean of the two floating-point values. For nominal attributes, we found in cross-validation experiments on the PAN'15 training datasets

that the two methods usually agreed. However, in cases where the predictions differed, our primary feature set model was correct more often except on the English-language subcorpora, where the secondary feature set seemed to slightly outperform the topic model features. We suspect this is due to some of the features in our secondary feature set being exclusively used for English data (as in the sentiment analysis features) or based on lexica containing mostly English words. Therefore, when making our final predictions for nominal attributes, we choose to accept the prediction made by the primary feature set model in Spanish, Italian and Dutch; in English, we use the nominal class label predicted by the secondary feature set model.

4 Results and Conclusion

The following table shows the prediction accuracy of our official entry to the PAN'15 Author Profiling task. The columns "Age," "Gender," and "Both" contain the fraction of authors classified correctly, while we list the RMSE for the personality attributes:

Language	**Global**	**Gender**	**Age**	**Both**	**RMSE**	**Agr.**	**Con.**	**Ext.**	**Open**	**Sta.**
English	0.67	0.73	0.72	0.51	0.16	0.15	0.14	0.15	0.15	0.22
Spanish	0.57	0.68	0.50	0.32	0.17	0.17	0.16	0.19	0.14	0.21
Italian	0.70	0.56	–	–	0.15	0.15	0.13	0.13	0.16	0.20
Dutch	0.84	0.81	–	–	0.14	0.15	0.14	0.15	0.09	0.17

We believe we have demonstrated that topic modeling is a promising direction for further research in prediction tasks such as author profiling. Our software achieved accuracy levels at or above average in most subtasks, among roughly 20 participating teams. We see possible avenues of improvement in the construction of our topic models through more informed selection of the LDA parameters, as well as the option of building multiple independent models with different starting conditions and combining the resulting predictions. As for our particular implementation choices, we might be able to improve our accuracy if we devoted more effort to optimizing the classifier training and testing, thus avoiding the need to use feature selection filters beyond what is inherent in the Bagging and RotationForest ensemble methods.

While WEKA was a useful experimental tool for trying different combinations of features and classifier settings, there was some overhead involved in formatting our data in WEKA compatible temporary files and calling the program with its required Java environment from own software written in Python. Now that we have a vision for a successful combination of features based on topic modeling, together with ensemble methods of classification, we plan to further refine these techniques and apply them to other prediction problems in the future.

References

1. Bamman, D., Eisenstein, J., Schnoebelen, T.: Gender identity and lexical variation in social media. Journal of Sociolinguistics 18(2), 135–160 (2014)

2. Bazo, A., Burghardt, M., Wolff, C.: Tworpus - an easy-to-use tool for the creation of tailored twitter corpora. In: Gurevych, I., Biemann, C., Zesch, T. (eds.) Language Processing and Knowledge in the Web, Lecture Notes in Computer Science, vol. 8105, pp. 23–34. Springer Berlin Heidelberg (2013), http://tools.mi.ur.de/tworpus/
3. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python. O'Reilly (2009)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. JMLR 3, 993–1022 (2003)
5. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)
6. Gollub, T., Stein, B., Burrows, S.: Ousting Ivory Tower Research: Towards a Web Framework for Providing Experiments as a Service. In: Hersh, B., Callan, J., Maarek, Y., Sanderson, M. (eds.) 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR 12). pp. 1125–1126. ACM (Aug 2012)
7. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: An update. SIGKDD Explor. Newsl. 11(1), 10–18 (Nov 2009)
8. Hutto, C., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: International AAAI Conference on Weblogs and Social Media (2014)
9. López-Monroy, A.P., Montes-y-Gómez, M., Escalante, H.J., Pineda, L.V.: Using intra-profile information for author profiling. In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) Working Notes for CLEF 2014 Conference, September 15-18, 2014. CEUR Workshop Proceedings, vol. 1180, pp. 1116–1120. CEUR-WS.org (2014)
10. McCallum, A.K.: Mallet: A machine learning for language toolkit (2002)
11. Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., Smith, N.A.: Improved part-of-speech tagging for online conversational text with word clusters. In: Vanderwende, L., III, H.D., Kirchhoff, K. (eds.) Proceedings of NAACL 2013. pp. 380–390. ACL (2013)
12. Pardo, F.M.R., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daelemans, W.: Overview of the author profiling task at PAN 2014. In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) Working Notes for CLEF 2014 Conference, September 15-18, 2014. CEUR Workshop Proceedings, vol. 1180, pp. 898–927. CEUR-WS.org (2014)
13. Rangel, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. In: Cappellato, L., Ferro, N., Jones, G., San Juan, E. (eds.) CLEF 2015 Labs and Workshops, Notebook Papers. CEUR-WS.org vol. 1391 (2015)
14. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010)
15. Rodriguez, J.J., Kuncheva, L.I., Alonso, C.J.: Rotation forest: A new classifier ensemble method. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(10), 1619–1630 (2006)
16. Schwartz, H.A., Eichstaedt, J.C., Kern, M.L., Dziurzynski, L., Ramones, S.M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M.E.P., Ungar, L.H.: Personality, gender, and age in the language of social media: The open-vocabulary approach. PLoS ONE 8(9) (09 2013)
17. Steyvers, M., Griffiths, T.: Probabilistic Topic Models. In: Landauer, T., McNamara, D., Dennis, S., Kintsch, W. (eds.) Latent Semantic Analysis: A Road to Meaning, chap. Probabilistic Topic Models. Lawrence Erlbaum Associates (2006)
18. Weren, E.R.D., Kauer, A.U., Mizusaki, L., Moreira, V.P., Palazzo Moreira de Oliveira, J., Wives, L.K.: Examining multiple features for author profiling. JIDM 5(3), 266–279 (2014)
19. Weren, E.R.D., Moreira, V.P., Palazzo M. de Oliveira, J.: Exploring information retrieval features for author profiling. In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) Working Notes for CLEF 2014 Conference, September 15-18, 2014. CEUR Workshop Proceedings, vol. 1180, pp. 1164–1171. CEUR-WS.org (2014)