Distribution Agreement

Mingyang Sun April 9, 2019

Towards Personality Trait Prediction from Chatbot Conversations Using Machine Learning with Domain Adaptation

By

Mingyang Sun

Eugene Agichtein, Ph.D
Adviser

Computer Science

Eugene Agichtein

Adviser

Joyce Ho, Ph.D

Committee Member

Phillip Wolff, Ph.D

Committee Member

2019

Towards Personality Trait Prediction from Chatbot Conversations Using Machine Learning with Domain Adaptation

By

Mingyang Sun

Eugene Agichtein

Adviser

An abstract of
a thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Sciences with Honors

Computer Science

2019

Abstract

Towards Personality Trait Prediction from Chatbot Conversations Using Machine Learning with Domain Adaptation

By Mingyang Sun

Accurate personality prediction has been proven to be useful for tasks like solving the cold-start problem in personalized recommendation[1]. In recent years, a number of research works have been published in different areas: written texts[2], movie scripts[3] and social media[4], with natural language processing (NLP) techniques and machine learning algorithms. In the field of open domain conversations, however, automatic personality trait detection has only been studies on natural human-human conversations, but not human-machine conversations. Under this circumstance, we present first study on personality trait prediction from open-domain conversations with a chatbot.

As intelligent assistants, such as Google Assistant, Apple Siri and Amazon Alexa, have gained increasing popularity with the development of mobile devices, the potential of usefulness of personality prediction on human-machine conversations data can be extensive. News recommendation function in these intelligent assistant systems, for example, can take users' personality as a reference: users with positive score on openness trait tend to be interested in aesthetic activities, so they possibly would like to know about trending news about new art shows, exhibitions and movies, while users with high consciousness might be attracted more by things happening in the White House. Therefore we believe detecting personality traits during conversations with users is a both challenging and valuable task.

In this thesis, we confirm the feasibility of user personality trait recognition in the open-domain human-machine conversations. We explore three methods: 1) models learned on engineered features, 2) models learned on transformed features mapped by linking functions constructed through heterogeneous domain adaptation, and 3) domain adaptation approaches applied to transformed features with social media data as the auxiliary task. The experimental results on real conversations with users support the feasibility off this task and suggest promising directions for future research.

Towards Personality Trait Prediction from Chatbot Conversations Using Machine Learning with Domain Adaptation

By

Mingyang Sun

Eugene Agichtein

Adviser

A thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Sciences with Honors

Computer Science

2019

Acknowledgements

First, thank you to all my committee members, Dr. Eugene Agichtein, Dr. Joyce Ho and Dr. Phillip Wolff, who have spared time to attend my honors thesis defense, for giving future directions, offering valuable advice and providing multiple innovative perspectives to me. This experience is, I believe, part of my most unforgettable moments in Emory University.

Second, I want to thank Alexa Prize for giving me the chance to participate in such a project and Emory IRLab for supporting me to be part of the competition as an undergraduate student. Also, thank all members in Emory IrisBot team for partly supporting this research, providing the data and inspiring me through meetings and discussions. I would like especially thank my teammate and friend Harshita Sahijwani for generously offering both research support as the data annotator and emotional support as a friend.

Third, I would like to give special thanks to my advisor Dr. Eugene Agichtein, who has guided me through my whole research experience in my undergraduate years. In 2018 I joined the Emory IRLab as a member of IrisBot team for Alexa Prize, where the whole research idea of this thesis comes from, and Dr. Agichtein has provided me with lots of deep material in information retrieval, machine learning and deep learning, with his profound knowledge background during the past two years. I wouldn't accomplish such a work without his guidance and help.

Last, I would like to thank my parents and all my friends for being the most awesome people in my world.

Table of Contents

List of Figures

List of Tables

April 24, 2019

Chapter 1 Introduction

1.1 Background and Motivation

Accurate personality prediction algorithms have been an active area of research, spanning across the fields of psychology, natural language processing, and machine learning. The ultimate goal is, generally, to provide better personalized service by profiling users. For example, Zhang and Zhao[1] proposed a recommendation model to solve the cold-start problem for new users with speech personality traits. Several other researches also examine the practicability of enhancing user profiling[5] and alleviating cold-start problem with collaborative filtering[6] by grouping people based on predicted personalities. During the past, a number of automatic personality trait prediction experiments have been conducted in social media field, but few in open-domain conversational systems.

Conversational AI designed for non-task oriented dialogues has been an active research area only recently[7] but with developments of natural language processing (NLP) and natural language understanding (NLU), intelligent assistants have increasingly satisfying performance over time and thus gained popularity. The improvement of personality prediction performance in the intelligent assistant system, based on achievements in social media field, could benefit numerous users as an auxiliary task to provide better service such as topic customization and high quality recommendation. For example, users with high openness score would prefer areas related to arts and music, and they might willing to know about the ticket information of recent concerts around. Hence, we are aspired to build highly accurate personality trait recognition models for open-domain dialogue systems.

Though lots of accomplished models have been proven to efficiently predict users' personalities on social media such as Facebook and Twitter[4], it remains a challenging task in open-domain conversational systems due to several reasons. First, the data patterns in conversational systems are different from all previous areas: most conversations have fewer turns than posts available under someone's Facebook account, users usually don't talk too much - most times they would tend to answer questions simply with "yes" or "no", and due to automatic recognition speech (ASR) errors, some conversations are even not understandable by human beings. The second problem is the limited number of available data. Most released dialogue datasets are either not human-machine conversations, like Cornell Movie-Dialogs Corpus[8], or task-oriented chats, like MultiWOZ Corpus[9], and even qualified ones like ConvAI datasets[10] lack personality labels.

1.2 Problem Statement

The problem is to predict users' polarity of personality traits from open-domain human-machine conversations. We also assume that we have a set of m labeled social media external data with binary labels (0 or 1) on c personality traits which could be used as the auxiliary task. We define the social media domain as the **SoMe** domain and the open-domain human-machine conversation domain as the **ODHMC** domain.

Given: a set of n open-domain human-machine conversations.
Output: n binary predictions (0 for negative and 1 for positive) on each of c personality traits.

The binary prediction is appropriate for the open-domain dialogue system, as we don't aim to predict the exact levels of users' certain personality trait, but expect to estimate the trait polarity to profile users.

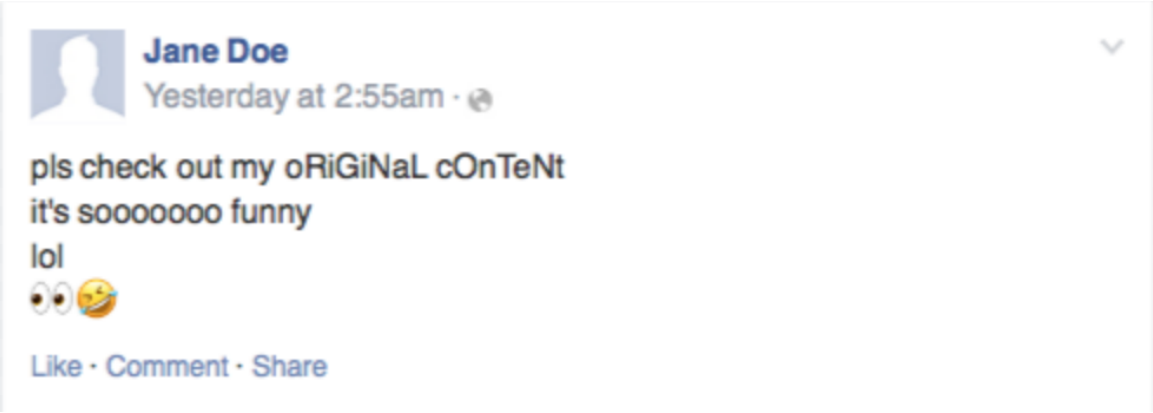
1.3 Research Overview

We base this research on the previous works in the social media field: users' personality traits prediction on collected information from their social media accounts. Examining all previous datasets carefully, we found shared commonalities between especially social media data and open-domain conversation data: they are self-expressive, composed of short fragmented texts, with less formal language cues and more semantically and syntactically meaningless forms. The sample data of an open-domain human-machine conversation and a social media post are illustrated in the Figure 1.1 and the Figure 1.2.

BOT: Would you like to hear about popular tracks this week?
USER: Okie
BOT: Here is one top recommendation from Billboard, titled God's Plan by Drake. Are you interested in hearing more about Drake?
USER: Nah, no Drake, some animal facts maybe.
BOT: We can talk about animals! I have some fun facts about them. What is your favorite animal?
USER: Love kung fu panda.

Figure 1.1: Replicated sample human machine conversation

Researches and data on the similar task in social media could be beneficial. On the one hand, features applicable to social media data should be helpful for the



Jane Doe
Yesterday at 2:55am ·

pls check out my oRiGiNaL cOnTeNt
it's sooooooo funny
lol
👀🤣

Figure 1.2: Replicated sample Facebook post

open-domain conversation system task. Our engineered feature set on conversation data include crucial features for social media task. On the other hand, as there are more labeled, collected social media data, like MyPersonality project from Facebook [11], these datasets could be used as a "source" domain, or an auxiliary task, to generalize model performance on the "target" domain, open-domain human-machine conversations. Therefore, besides comparing machine learning approaches on original conversation features, we also apply domain adaptation techniques to transform features and evaluate the influence of using social media data as a source domain on the personality traits prediction task in open-domain human-machine conversation area.

1.4 Proposed Methods

Our proposed methods can be summarized into two stages: Stage 1) we construct feature matrix respectively for the SoMe domain and the ODHMC domain. Stage 2) we apply a set of domain adaptation techniques to transform both domains into a common domain, where we develop a number of generalized models. The details of the whole methodology is illustrated in the Chapter 4.

1.4.1 Feature Matrix Construction

We extract different feature sets for SoMe domain and ODHMC domain. These two feature sets share a list of overlapped features yet also include domain-specific features. Common feature sets used in social media prediction task include Linguistic Inquiry and Word Count (LIWC)[12] features, word embedding features, vocabulary features[13]. Prediction models built up on heterogeneous information make use of other non-text features, such as profile pictures, images posted, profile information, and interaction patterns[14]. Word-embedding features have been proven to be amazingly powerful in the social media prediction task[15]. Open-domain human-machine conversation data, however, only contain utterances, and users' answers tend to be straightforward and short, i.e., the ability of pure text features is expected to be very limited. Thus we assume that responsive-pattern features[16] and hand-crafted interaction features would play some significant roles.

1.4.2 Domain Adaptation

The incorporation of domain adaptation allows to combine data from both SoMe domain and ODHMC domain together by mapping all data onto a common domain space. In this case, the generated models from the combined data will be generalized to catch signals from both domains. In this thesis, all domain adaptation techniques are implemented under a supervised setting, in which all data are labeled already.

- **Simple Feature Augmentation.** Given K domains with the same feature set, this approach simply augments the feature space for each domain by making $K+1$ copies of original features. Each copy represents a domain (for example, the weight of each feature of copy $Copy_A$ represents the feature weight in the domain A), and the one extra copy represents a general version. It is first proposed by Daumé (2009)[17]. This simple feature augmentation, however, requires that the features of multiple domains should be on the same space. As we construct different feature matrices for the domains in our problem, we use a manifold alignment approach, heterogeneous domain adaptation, to transform both original feature sets onto a latent common feature space and then apply this domain adaptation method.

- **Heterogeneous Domain Adaptation (HDA).** The manifold alignment based heterogeneous domain adaptation we use for this research is adapted from the work by Wang and Mahadevan[18]. The algorithm creates linking functions to map different features from various domain onto the same feature sub-space by aligning labels. On the latent feature sub-space, this approach works by bringing instances from the same class (from the same or different domains) together while separating instances from different classes as far as possible. We make use of an adapted version, as it's initially aimed for semi-supervised settings, to map SoMe domain and ODHMC domain data onto a common feature space, so they can be combined to be used for simple feature augmentation above and the stacked denoising autoencoders below.

- **Stacked Denoising Autoencoders (sDAE).** By reconstructing input through encoder and decoder functions, autoencoders are powerful in generating compressed data while preserving the original distribution. The autoencoder can be stacked on top of each other to take the output from the previous layer as the input, thus forming a stacked autoencoders. Denoising autoencoder is an alternative for the regular autoencoder as it prevents learning an identity matrix by randomly turning input vectors to be 0. We use the sDAE to learn different data representations from the transformed features and do prediction on this new dimension.

1.5 Contributions

According to our knowledge, this is the first work to predict personality from open-domain human-machine conversations. The contributions of this thesis are threefold:

- We successfully confirm the feasibility to predict users' personality traits from open-domain human-machine conversations.

- We design a set of novel features to capture personality trait information for open-domain human-machine conversations (Section 4.1).

- We present a series of principles and experiments results with domain adaptation (DA) approaches and multi-task learning (MTL) techniques (Section 4.2, 4.3 4.4 and Section 5.2).

We believe with the advanced development of intelligent assistants, incorporating personal traits including personalities into the end-to-end open-domain conversation systems would be crucial to provide better personalized services, such as customized topic suggestions, engaging conversation flows, and accurate news, events and items recommendations.

This project provides several basic ideas for advanced personality prediction task in the future. More advanced works could incorporate other heterogeneous information like prosody features or user profile information obtained through side channels.

Chapter 2 Related Work

2.1 Overview: Personality in Psychology

Though a number of popular models of personality traits are used for different research purposes, the Five Factor Model (the "Big Five") validated by McCrae and Costa Jr (1970s) is the most recognized model today. A brief description of the Big Five traits is provided by Rossberger below:

- **Opennes to Experience:** extent to which individuals exhibit intellectual curiosity, self-awareness, and individualism/nonconformance.
- **Conscientiousness:** extent to which individuals value planning, possess the quality of persistence, and are achievement-oriented.
- **Extraversion:** extent to which individuals exhibit intellectual curiosity, self-awareness, and individualism/nonconformance.
- **Agreeableness:** extent to which individuals value cooperation and social harmony, honesty, decency, and trustworthiness. Agreeable individuals also tend to have an optimistic view of human nature.
- **Neuroticism:** extent to which individuals experience negative feelings and their tendency to emotionally overreact.

These five personality traits have been obtained repeatedly by applying factor analysis to trait adjectives used in personality questionnaires (Norman, 1963)[19], of which the basis is the Lexical Hypothesis (Allport Odbert, 1936)[20], i.e. the most relevant individual differences are encoded in the words.

Experiments using the Big Five in the psychology field have indicated that personalities influence many aspects of personal behavior, such as leadership abilities(Hogan, Curphy, Hogan, 1994)[21] and academic ability and motivation[22]. For example, according to Marshall et al.[23], people with high extroversion scores tend to talk more about social activities/everyday life, and people with high openness scores would prefer to share their opinions on intellectual events and achievements.

Till around 2006, most personality-related researches continue exploring on the relationships between personalities and various aspects of social lives, but there were still a little work on the automatic recognition of personality traits (Oberlander Nowson, 2006)[24]. Then with the development of data analysis and computational power, the focus started to switch to the automatic personality assessment through heterogeneous methods.

2.2 Personality Prediction in Texts

Assessments on texts originate from the fundamental idea of factor analysis: the Lexical Hypothesis.

Pennebaker and King (1999)[25] report a summary of linguistic features associated with each of the Big Five personalities by using Linguistic Inquiry and Word Count (LIWC) tool to count word categories of 2,479 essays written by psychology students whose personalities have been assessed with a questionnaire. The authors found a number of fragmented but significant correlations between linguistic dimensions and personalities. For example, "openness" is characterized by a tendency to use longer and tentative words (e.g., perhaps), as well as the avoidance of 1st person singular pronouns and present tense forms.

Inspired by their work, Mairesse et al. (2007)[26] present the first work on exploring the use of classification models, regression models and ranking models with different feature sets on data collected by Pennebaker and King (1999)[25]. The general feature set they used includes both syntactic(e.g., ratio of pronouns) and semantic information (e.g., positive emotion words). Besides, they also add 14 additional features from the MRC Psycholinguistic database, which contains statistics for over 150,000 words, such as estimates of the age of acquisition and familiarity. All models significantly outperform the baseline. Models with LIWC feature set, particularly, improve the most. The results strongly confirm their assumption that personality can be recognized by computers through language cues.

Luyckx and Daelemans (2008) published Personae, a new corpus consisting of 145 student essays of 1400 words on average for computational stylometry, especially on authorship attribution and the prediction of author personality from text[27]. They also creatively incorporated meta-information, like personal profiles of the authors, associated with texts to help. The paper concludes that, using combinations of good working lexical and syntactic features, exploratory experiments, introverted-extroverted traits can be predicted fairly accurately, with the accuracy score of 64.14% (introversion vs. non-introversion) and 60.00% (extroversion vs. non-extroversion) separately.

Built upon their work, an improved system by Noecker et al. (2013)[28] compares the document model, choosing the closest neighbors by the cosine distance metric on the feature vector, and the centroid model, comparing the document with the centroid for each personality category. Their work confirms both the usefulness of simple character-level feature sets like character tetragrams and the efficiency of normalized dot-product nearest neighbor classifier.

2.3 Personality Prediction in Social Media

Compared with long, well-formatted essays, researchers are obviously more interested in detecting personality traits on social media texts, because of its larger potential to be used for personalized retrieval and recommender system[6].

One of the earliest works was published by Golbeck et al.[29], in which a variety of language features and social network behavior features are used to predict user personalities. They collect a simple set of statistics about each user's twitter accounts to represent their "Twitter Use" features, including Number of Followers, Number of Following, Number of "@Mentions", Number of Links and Number of Hash Tags. This research proposes a prototype for the following works: a combination of text and non-text features could help to improve the prediction models in social media. Admitted by authors, one trait is that the analysis relies heavily on texts, yet they extract text-related features simply by LIWC and MRC tools, as mentioned by Mairesse et al. (2007)[26]. Besides text features, researchers begin incorporating other heterogeneous features like *Likes* and profile information in these tasks. Skowron et al.[14] presents their work on fusing social media cues from both Twitter and Facebook, and proposes the usage of social media specific features like number of follower and followees.

One of the most accomplished work by Wei et al.[16] introduces a very systematic methodology in prediction tasks. Also, not using closed-vocabulary approaches anymore, this work begins to adopt open-vocabulary approach by modifying the idea from Kim[30]: they use a modified version of text-cnn to encode tweets as the language presentation. They also use innovate features including emoticons and avatars. In the final stage their framework has a stacked generalization-based ensemble layer to predict the final result.

2.4 Personality Prediction in Conversations

Automated personality trait prediction from speech and conversations also attracts some attention. Mairess et al. (2007)[26] publish their experiment results on Electronically Activated Recorder (EAR) dataset, collected by Mehl et al.[31]. Besides text features they include for prediction on essay dataset, they make use of audio features including voice pitch, voiced time, and speech rate. They find that prosody features are especially helpful for identifying extroverts and open-minded individuals.

With the widespread of intelligent assistants, human-machine conversation gains increasing attention. According to our knowledge, our work is the first to predict personality traits from open-domain human-machine conversations. In the paper from Fang et al. (2018)[32] which describes their design for SoundingBoard, an Alexa Prize social bot, they mention the influence of users' personalities on interaction with the bots. They conduct the experiment by asking users to take a short personality survey at the beginning of the conversation, in order to somehow guide the conversation topic selection, and classify users into 4 types with 2 traits (openness and extraversion) according to the quiz results. They calculate pearson correlation between each trait and some interaction criteria. The results show that extraversion has statistically significant positive correlations with high ratings, number of turns in conversations and average utterance length.

Some other works focus on users' personal traits prediction rather than personalities. Tigunova et al.[33] devise a neural architecture, the *Hidden Attribute Model (HAM)*, trained with subject-predicate- object triples, to predict personal attributes of subjects. Their approach can successfully infer that a subject who uses terms like *theory*, *mathematical* and *species* tend to be a *scientist*.

This research differs from all previous works. First, our study is the first to develop personality traits prediction models on open-domain human-machine conversations. Second, we incorporate external domain data by domain adaptation techniques to discover shared latent space between domains, and generate more generalized model on common feature space. We present a systematic view of all experiment results and discuss the lessons of using other data as auxiliary domain in this task.

Chapter 3 Datasets

3.1 MyPersonality Project Dataset

MyPersonality[11] is an online application that can be installed through Facebook. The application offers the user a detailed explanation of his/her personality trats once he/she fulfills proposed questionnaires. According to the developers they have collected data from more that 5 million users. Part of these data is publicly available: we take this publicly available subset of 250 users with around 9,917 status shared. This dataset also comes with numerical personality score on a basis of 1 to 5, and a binary indicator (y/n) based on the score. The posts it contains are original so they have all the characteristics of social media texts: not syntactically meaningful, lots of abbreviations and random words and slang, and even in some cases full of emoticons.

This dataset contains 176 users with 'y' (i.e. 1) on openness trait, accounting for 70.4% of the samples, and 96 users with 'y' on extraversion trait, accounting for 38.4%. Hence, on either dimension the data class distribution is somehow biased. Each user has 39.7 posts on average, and the number of posts varies largely by users, as the standard deviation of count of posts is as large as 43.5. Each post has 14.5 words on average, and the distribution of word counts also spreads as the standard deviation of post word count is 12.5 overall.

3.2 Alexa Prize Chatbot Conversation Dataset

Alexa Prize 2018, sponsored by Amazon, is a worldwide research competition on building up advanced conversational social bots. Our conversation system talked with thousands of Alexa users from different areas during the contest on mainly 14 common topics, including music, movies, animals, sports, news, etc. The conversations used for this study come from a stable version in the final stage of semifinal, and we select them based on the following criteria:

- **Meaningful to human beings.** Some conversations are obviously not "meaningful", either because of automatic speech recognition (ASR) errors or users' intention. For example, some conversations are full of broken, non-meaningful words, might be the result of users' accents. Also some users intentionally break the conversation flows: we have some of them who keep cursing everything or just begin reading lots of lyrics.

- **At least 10 turns.** Conversations too short are ruled out because opening stage and stop command already take up 4 turns, and almost no information can be obtained if the conversation length is less than 10 turns.

The selected data has 22.84 turns on average, with the standard deviation of 13.1. Users tend to reveal much fewer linguistic cues by potentially talking little to the bot, usually answering "yes" or "no" or using simple words like "next" and "change topic". Compared to the word counts of posts in social media, each utterance has 2.6 words on average, with the standard deviation of 2.7. From these statistics we can see how different the formats of social media data and open-domain human-machine conversation data are.

We then manually annotate 180 conversations from selected ones. Continuing the idea from Fang et al.[32], we only label 2 traits, extraversion and openness, with 1 as positive examples and 0 as negative examples, by observing users' communication habits and the topics they are interested in.

We choose these 2 traits because 1) the data selection process is biased. Ruling out instances with intentionally breakage and full of profanity and cursing implicitly removes users with negative agreeableness polarity and positive neuroticism polarity. 2) they represent different aspects of users' interaction with the bot and are more easily to be manually annotated. Extraversion is represented through communication habits, (eg. more questions, tend to talk in long sentences, willingness to talk more) and openness is seen through topics users are interested in (eg. art, music, books, etc.) and users' willingness to accept new topics. 3) Consciousness can not be annotated even on self-agreement basis as the information we could obtain through conversations is noticeably little. It's not a trait easily told from either ways or content of conversations.

In the annotated dataset, there are 99 users with score '1' in openness trait, accounting for 55% of samples, and 96 users with score '1' in extraversion trait, accounting for 53%. Compared to the MyPersonality data, the conversation dataset has more balanced polarity distribution on each trait. To validate the manual labels, we also calculate the inter-annotator agreement on the annotation: out of 30 subset, we get 90% agreement on openness (27 out of 30) and 83.3% agreement on extraversion (25 out of 30).

Chapter 4 Methodology

We will present our methodology in this section. In Section 4.1 we describe the feature engineering techniques we design for the general setting (Section 4.1.1), the social media setting (Section 4.1.2) and the conversation setting (Section 4.1.3). In Section 4.2 we present the details of our multiple domain adaptation approaches, including Simple Feature Augmentation (Section 4.2.1), Heterogeneous Domain Adaptation (Section 4.2.2) and Stacked Denoising Autoencoders (Section 4.2.3).

4.1 Feature Engineering

In this study, features used for Mypersonality data and Alexa Conversation data are slightly different, but we adopt the same high-level architecture to generate the m x p feature matrix for both datasets in three general steps: 1) *Preprocessing.* To reduce the noise, texts are cleaned up by removing non-ascii and special characters, removing stopwords, lowering all cases, and removing numbers which have very little influence in distinguishing personalities[16]. 2) *Parallel Feature Extraction.* Processed data are fed into a list of parallel feature extractors, which will be explained in details in the Section 4.1.1, 4.1.2 and 4.1.3. There are roughly two types of features: text and non-text. 3) *Normalization.* Extracted features are first concatenated to be the m x p feature matrix, in which each row is the vector representation for a sample, and then the matrix is scaled to unit variance.

To keep the two domains similar, there is a number of common features shared by both datasets, introduced in the Secion 4.1.1. Features specific to each domain will be described in the Section 4.1.2 (Mypersonality) and Section 4.1.3 (Alexa).

The overall features we designed for this task roughly fall into four groups:

- **Text-related features**: Features extracted with Text-CNN architecture[30], bag-of-word clustering features and highly-correlated vocab features. (Described in 4.1.1.A, 4.1.1.B and 4.1.1.C).

- **Sentiment features**: Features extracted with Textblob Sentiment[34]. (Described in 4.1.1.D).

- **Responsive-pattern features**: Features extracted with modified version of Responsive-pattern CNN[16]. (Described in 4.1.3.A).

- **Interaction and Behavioral Pattern features**: Features hand-crafted based on previous works in the social media field. (Described in 4.1.2 and 4.1.3.B).

4.1.1 General Feature Engineering

In this section, we introduce a list of shared features between social media data and conversation data. The commonalities between two datasets have been discussed thoroughly in previous sections, so we assume that features work for social media task should perform well on conversation data also. The strategies used to extract common features, inspired by the work from Wei et al.[16], include Text-CNN, Bag-of-Words Clustering, Pearson Correlation and Sentiment Analysis. **All the features introduced in this section are extracted separately from both SoMe domain and ODHMC domain.**

A. Text-CNN

Instead of closed-vocabulary approaches like LIWC or MRC, we decide to use the open-vocabulary approach as it's fast, flexible, and especially fit for short-text tasks [15]. Deep learning has been proven to powerfully learn vector representation in several NLP tasks (Kim, 2014) [30]. We adopt a modified graphic model of a convolutional network structure (Kim, 2014) [30] because of its ability to model the sequential dependency of a sentence. The standard CNN has limitations in text understanding as no direct correlation exists among adjacent dimensions of embedded word vectors. Therefore, the modified structure applies convolution and max-pooling operation with different kernel sizes. In this case we discard convolutional layers and use the concatenated max-pooling results as the input for the next phase directly [30].

We use GloVe embeddings trained on Twitter [35]. Each word is represented by a 50-dimension vector, and each sentence of length n is represented as an n x 50 matrix. We then apply k kernels with various sizes to convolute embeddings. The feature vector learned by kernel j is of size s denoted as

$$\boldsymbol{c_j} = \left(c_{\mathrm{1,\ s}},\ c_{\mathrm{s+1,\ 2s}},\ c_{\mathrm{2s+1,\ 3s}},\ \ldots,\ c_{\mathrm{n\text{-}s+1,\ n}}\right) \tag{4.1}$$

The mapped feature result of kernel j is obtained by the max-pooling operation on this feature vector

$$\boldsymbol{\hat{c}_j} = \textit{max-pooling}\left(\boldsymbol{c_j}\right) \tag{4.2}$$

All max-pooled results of kernels will be concatenated to form the input for the next phase

$$\boldsymbol{C_{input}} = \boldsymbol{c_1} + \boldsymbol{c_2} + \boldsymbol{c_3} + \ldots + \boldsymbol{c_k} \tag{4.3}$$

The detailed structure of our Text-CNN is shown in the Figure 4.1. We use filters of size 2, 3, 4 and 2 filters for each size. The concatenated max-pooling result is used for the prediction layer.

The Text-CNN features for each user are calculated by the number of sentences they have for the polarities of each trait: op_1 and op_0 represent the number of sentences classified as "openness = 1" and "openness = 0" respectively; ex_1 and ex_0 represent the number of sentences classified as "extraversion = 1" and "extraversion = 0" respectively. **B. Bag-of-Words Clustering**

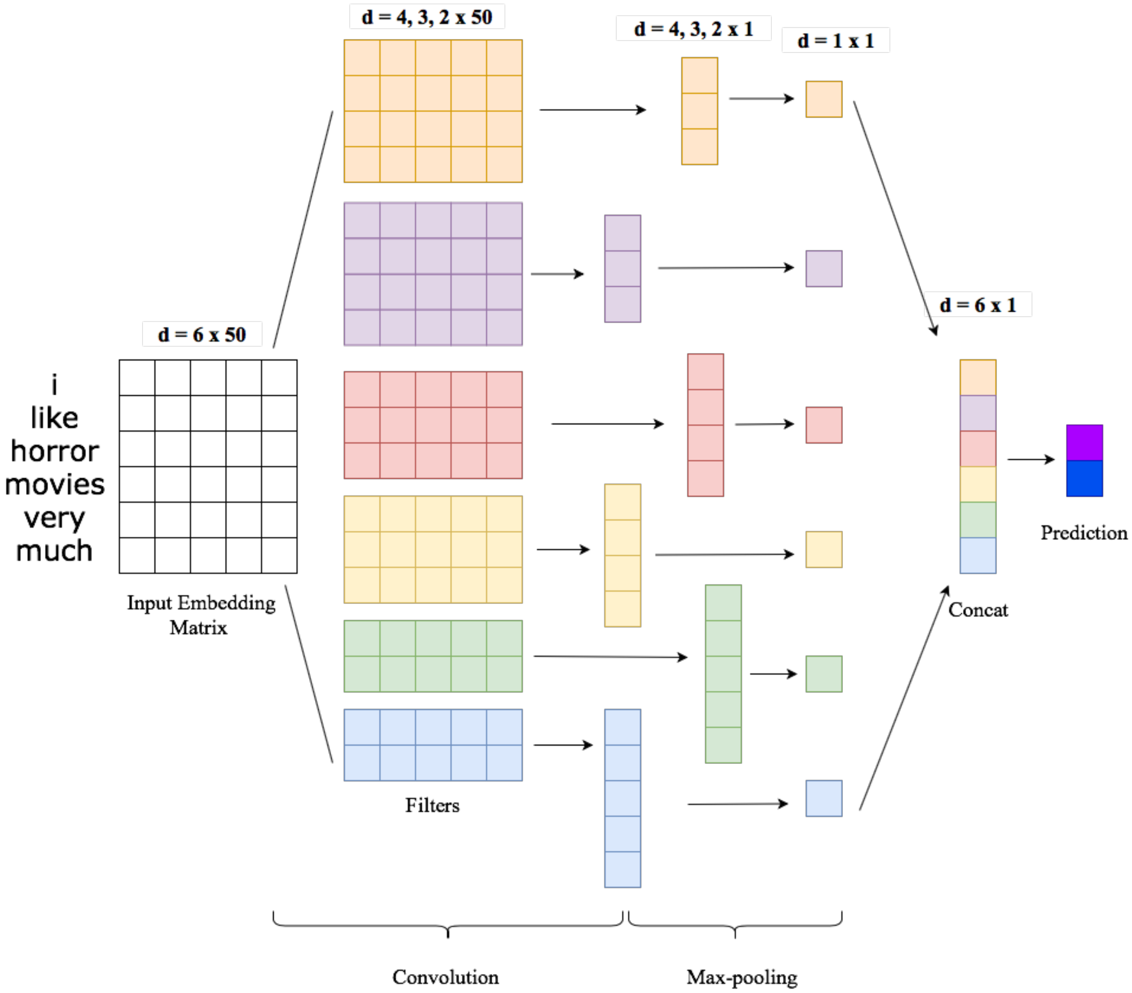


Figure 4.1: Illustration of the Text-CNN structure.

For the general representation of a user, i.e. all posts/utterances the user has, LIWC feature vector cannot be a good choice because of apparent sparsity as the word vocabularies for both datasets are limited. Hence we use the cluster-format vector instead. To reduce the noise, we calculate the top 75% frequent words and use k-means algorithm to cluster words by word vectors from the same GloVe model [35] we use for Text-CNN. The less frequent words then are mostly specific entities. . We plot several values of number of clusters and find out that 10 is the elbow point. Then each user's bag of posts/utterances is represented as a 10-dimension vector of count of words under each cluster.

C. Pearson Correlation

This extractor simply calculates the Pearson correlation between each word and the target personality dimension and selects top k and bottom k ($k = 50$ for social media data and $k = 20$ for conversation data, words in between are too specific entities to use) words with highest and lowest scores separately.

opn_pos_count	the count of words with top k positive PCC scores to openness
opn_neg_count	the count of words with top k negative PCC scores to openness
opn_pos_score	the sum of scores of all words with top k positive PCC scores to openness
opn_neg_score	the sum of scores of all words with top k negative PCC scores to openness
ext_pos_count	the count of words with top k positive PCC scores to extraversion
ext_neg_count	the count of words with top k negative PCC scores to extraversion
ext_pos_score	the sum of scores of all words with top k positive PCC scores to extraversion
ext_neg_score	the sum of scores of all words with top k negative PCC scores to extraversion

Table 4.1: Pearson Correlation Features on Both Traits.

D. Sentiment Analysis

We choose to use TextBlob[34] to calculate sentiment scores. The hand-crafted sentiment feature set is described in the following table. Sentiment-related features have been shown to improve recognition of linguistic constraints, thus be helpful for better classification [29].

avg_senti_score	average sentiment score over all sentences
std_senti_score	standard deviation of sentiment scores over all sentences
senti_pos_score	average positive sentiment score over positive sentences
senti_neg_score	average negative sentiment score over negative sentences
senti_pos_count	total count of all positive sentences
senti_neg_count	total count of all negative sentences
senti_neu_score	total count of all neutral sentences
senti_trans_score	total count of sentiment changes (eg. pos -> neg)

Table 4.2: sentiment-related features.

4.1.2 Specific Feature Engineering on Social Media Dataset

Features specific to social network data are characterized by interactions, with friends or with trending topics. These features are part of the feature set used in previous prediction works on social media data [14]. As MyPersonality project doesn't include users' profile information, such as number of friends, gender, demographic information, or avatar, there are very few features except those based on pure texts.

URLs	The normalized count of URLs. The content of URLs is categorized by vocabularies of pre-defined topics[36]. For example, if a post contains url: www.billboard.com/charts/Hot-100, then the count under topic "music" will increase by 1.
@Mentions	The normalized count of @Mentions and the diversity of @Mentions, i.e. the count of unique ones.
#Hashtags	The normalized count of #Hashtags and the content is categorized by vocabularies of pre-defined topics.

Table 4.3: Hand-crafted SoMe Domain Specific Features.

4.1.3 Specific Feature Engineering on Open-domain Human-machine Conversation Dataset

As we've shown the data statistics in Section 3.2, we expect the predictive ability of linguistic features would be limited on conversation data. Hence, we model users' interaction and behavior with two specific sets for the conversation dataset: 1) *Responsive-CNN features*, features extracted by the Responsive-CNN structure[16] to model the interaction between the bot and the user and 2) *Behavioral Pattern features*, manually crafted features representing behavioral patterns of users, reflecting their communication habits and interested content.

A. Responsive-CNN

According to psychology theories, different reactions to the same scenarios reflect people's different personalities. In this case, we use a modified version of the Responsive-CNN structure proposed by Wei et al[16] to encode users' interaction with the bot. This model is essentially an adapted version of Text-CNN structure. In the Text-CNN, each row vector represents a word, and if we use the row vector to represent a sentence, then the operations on row vectors could be regarded as modeling the interactions. To embed each utterance, either by the bot or the user, we use a pre-trained Doc2Vec[37] model to infer sentence vectors. The embedding vectors form the "conversation matrix", the input layer of Responsive-CNN. The structure of Responsive-CNN is exactly the same with that of Text-CNN introduced in the Section 4.1.1.A, and the only difference is that instead of using word vectors to form a sentence matrix, now the input is a conversation matrix consisted of sentence embeddings (i.e. each vector represents a sentence). As we've noticed that users in open-domain human-machine conversations normally tend to speak less, we assume the power of text-related features would be very limited, and interaction features should be emphasized to better learn users' personality differences.

B. Behavioral Patterns

We believe users' personalities could be reflected through their interactions, and we also assume that users' non-text-based behavioral patterns related to conversations are also crucial to identifying users' personality traits. It makes sense in daily scenarios: if we meet someone who'd show interests in most music and art topics we talk about, we'd like to classify this person as "open-minded" (i.e. high openness scores). Therefore, we add a set of manually crafted heterogeneous non-text features to represent users' behavioral patterns.

num_of_turns	the number of turns in this conversation
wc_avg	average word count of each utterance
wc_std	measure the variation of the length of utterance
yes_ratio	the ratio of utterance that expresses "yes"
no_ratio	the ratio of utterance that expresses "no"
suggested_topic_cov	the number of topics suggested by the bot
states_cov	the number of topics discussed by the user and the bot
state_min_engagement	the min number of turns of staying in one topic (eg. keep talking about music)
state_max_engagement	the max number of turns of staying in one topic
state_avg_engagement	the average number of turns of staying in one topic
state_std_engagement	measure the variance of users' topic stickiness

Table 4.4: Hand-crafted ODHMC Domain Specific Features.

These manually crafted features are designed based on two principles: 1) encoding users' ways of communication, these include number of turns in the conversation and information related to the word count, and 2) encoding users' attitude towards topics, these include information about suggested topics, stickiness to certain topics and the topics the bot and the users actually discuss about.

4.2 Domain Adaptation

Domain adaptation technique arises when we are *transferring knowledge* from the source domain to the target domain and *generalizing the model*, also referred as transfer learning. It's proven to be especially useful in two scenarios: 1) the target domain data is potentially biased (for example, the target domain is the blood pressure of all population but we have only collected data from senior people) and 2) the target domain data is limited (for example, we have a refined sentiment classification model for customer reviews on books and now we'd like to learn a model for hotel reviews). This is an appropriate approach for our situation. Now the challenge is to transfer domains so the classifier trained with the help of source domain could have a satisfying performance on the target domain.

In this section, we demonstrate 3 domain adaptation approaches we use in this study: Simple Feature Augmentation, Heterogeneous Domain Adaptation (HDA) and Stacked Denoising Autoencoders (sDAE). The domain adaptation allows us to transfer both SoMe domain and ODHMC domain onto the same domain, i.e., feature space, so we can use training data from social media as the auxiliary to classify personality traits in conversations. This helps us to construct more generalized models to capture signals from both domains. For each method, we will first discuss the general ideas behind it, and then explain in details how we adopt it specifically for our experiments.

4.2.1 Simple Feature Augmentation

We expand the data by augmenting feature space, with the approach proposed by Daumé (2009)[17], named Frustratingly Easy Domain Adaptation (FEDA) approach. It has been proven to be successful in several NLP tasks including entity recognition and review sentiment classification. We make use of this idea to explore the significance of a non-deep-learning domain adaptation. The advantages of FEDA algorithm are simple interpretability and implementation.

The algorithm basically takes each feature in the original problem and makes three versions of it: a general version, a source-specific version and a target-specific version. Then the augmented source data will contain only general version and source-specific version and the augmented target data will contain general version and target specific version, as shown in the following equation, where $\mathbf{0}$ is simply the zero vector. In one word, for K domains, the augmented feature space will contain $K+1$ copies of the original feature space.

$$\Phi^s(\mathbf{x}) =< \mathbf{x}, \mathbf{x}, \mathbf{0} >, \ \Phi^t(\mathbf{x}) =< \mathbf{x}, \mathbf{0}, \mathbf{x} > \tag{4.4}$$

The FEDA algorithm captures feature differences in various space by distributing weights. For example, to train a sentiment classifier in hotel (target) domains with smartphone (source) domains, this algorithm is able to capture three types of features by deciding their weights on different spaces: 'excellent' is a shared feature for positive review so it has positive weight in shared space, 'small' has opposite meanings in two spaces so it has positive weight in smartphone space while negative weight in hotel space, and 'lounge' is only existent in the hotel domain so its weight is captured in the target feature space only.

We expect that this algorithm is able to emphasize commonalities and also capture the differences in the latent feature subspace of the SoMe domain and the ODHMC domain. One obstacle is that we have two similar but different feature sets for these two domains but we need joint, common feature space, so we first transform both original feature sets with the HDA algorithm introduced in the Section 4.2.2, and then perform this feature augmentation method on mapped features.

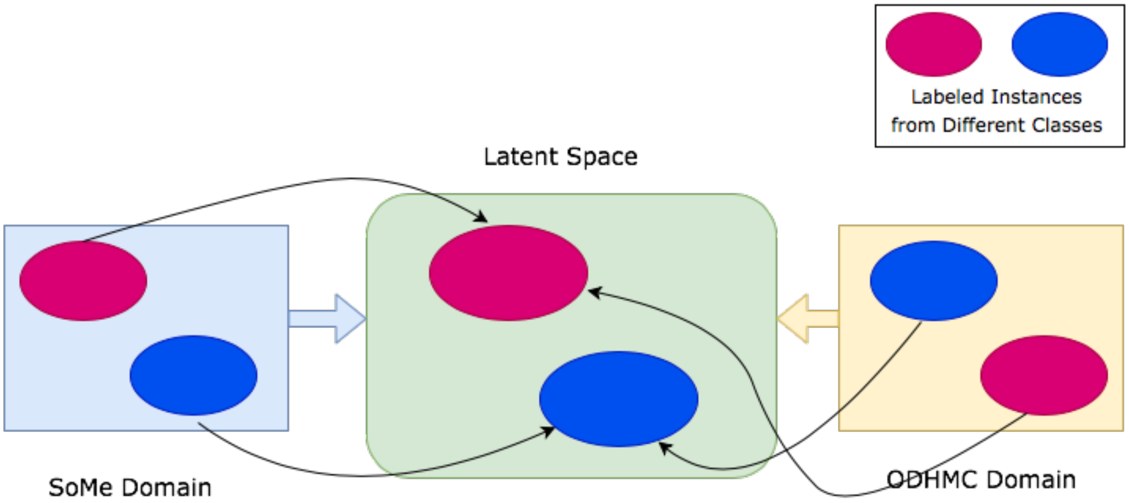


Figure 4.2: Illustration of adapted HDA with manifold using labels.

4.2.2 Heterogeneous Domain Adaptation

We make use of heterogeneous domain adaptation to link both feature spaces with mapping functions, and transform both feature sets into the common latent subspace.

In this study, we implement a modified manifold alignment based approach for heterogeneous domain adaptation proposed by Wang and Mahadevan[18], which extends the existing techniques by making use of labels rather than correspondences to align the manifolds. Therefore this approach doesn't rely on correspondence between domains but captures similarity through label information, and can be broadly applied to semi-supervised settings. As our experiments are carried out under supervised settings, the general idea of our modified version is illustrated in the Figure 4.2.

The goal of HDA is to construct mapping functions to map input sets into a new d dimensional space such that (1) the topology of each domain is preserved, (2) instances from the same class (i.e. the same label) are brought as close as possible together on the new latent space and (3) instances from the different class are separated as far as possible on the new space. Consequently, we are formulating a cost function to find the equilibrium to satisfy these three conditions as much as possible.

Before jumping to the cost function construction, the weight matrices have to be defined first. All matrices defined below are modified from the metrices by Wang and Mahadevan [18] and adapted to our setting. We take SoMe domain as domain X_1 with m instances and p_1 features and ODHMC domain as domain X_2 with n instances and p_2. features.

- **Similarity matrix** $W_\mathrm{s} = \begin{pmatrix} {W_\mathrm{s}}^{1,1} & {W_\mathrm{s}}^{1,2} \\ {W_\mathrm{s}}^{2,1} & {W_\mathrm{s}}^{2,2} \end{pmatrix}$. The similarity matrix W_s is a $(m{+}n)$ x $(m{+}n)$ matrix, where ${W_\mathrm{s}}^{a,b}(i,\ j) = 1$ if i and j are from the same class, otherwise 0. Then the corresponding diagonal row matrix $D_\mathrm{s}(i,\ i) = \sum_j W_\mathrm{s}(i,j)$, and the combinatorial graph Laplacian matrix $L_\mathrm{s} = D_\mathrm{s}$ - W_s.

- **Disimilarity matrix** $W_\mathrm{d} = \begin{pmatrix} {W_\mathrm{d}}^{1,1} & {W_\mathrm{d}}^{1,2} \\ {W_\mathrm{d}}^{2,1} & {W_\mathrm{d}}^{2,2} \end{pmatrix}$. The similarity matrix W_d is a $(m{+}n)$ x $(m{+}n)$ matrix, where ${W_\mathrm{d}}^{a,b}(i,\ j) = 1$ if i and j are from the different classes, otherwise 0. Then the corresponding diagonal row matrix $D_\mathrm{d}(i,\ i) = \sum_j W_\mathrm{d}(i,j)$, and the combinatorial graph Laplacian matrix $L_\mathrm{d} = D_\mathrm{d}$ - W_d.

- **Formulation of matrix $\boldsymbol{W}_\mathbf{k}$, $\boldsymbol{L}_\mathbf{k}$, $\boldsymbol{D}_\mathbf{k}$.** $W_\mathrm{k}(i,\ j)$ represents the within-domain similarity of instance i and instance j. The constructions of L_k and D_k are the same as above.

- **Matrices $\boldsymbol{L}$ and $\boldsymbol{Z}$.** $L = \begin{pmatrix} L_1 & 0 \\ 0 & L_2 \end{pmatrix}$ is a $(m{+}n)$ x $(m{+}n)$ matrix. $Z = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix}$ is a $(p_1{+}p_2)$ x $(m{+}n)$ matrix. These two matrices are used to model the input sets.

With weight matrices above defined, now we can construct the cost functions step by step based on the conditions. In the following equations, f_a and f_b are mapping functions.

- **Goal 1:** the instances from the same class are mapped to similar locations.
  $A = 0.5\sum_{a=1}^{2}\sum_{b=1}^{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\left\| {f_\mathrm{a}}^T {x_\mathrm{a}}^i - {f_\mathrm{b}}^T {x_\mathrm{b}}^j \right\|^2 W_\mathrm{s}(i,j)^{a,b}$

If ${x_\mathrm{a}}^i$ and ${x_\mathrm{b}}^j$ are from the same class, and their embeddings are far, then A will be large. Minimizing A will project instances from the same class to be close.

- **Goal 2:** the instances from different classes are well-separated from each other.

  $B = 0.5\sum_{a=1}^{2} \sum_{b=1}^{2} \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| {f_\mathrm{a}}^T {x_\mathrm{a}}^i - {f_\mathrm{b}}^T {x_\mathrm{b}}^j \right\|^2 W_\mathrm{d}(i,j)^{a,b}$

  If ${x_\mathrm{a}}^i$ and ${x_\mathrm{b}}^j$ are from the different classes, and their embeddings are close, then B will be small. Maximizing B will separate instances from different classes on the latent space.

- **Goal 3:** the topology of each set is preserved.

  $C = 0.5\mu \sum_{k=1}^{2} \sum_{i=1}^{m_\mathrm{k}} \sum_{j=1}^{m_\mathrm{k}} \left\| {f_\mathrm{k}}^T {x_\mathrm{k}}^i - {f_\mathrm{k}}^T {x_\mathrm{k}}^j \right\|^2 W_\mathrm{k}(i,j)$, where when k $=$ 1, m_k $=$ m, when k $=$2, m_k $=$ n.

  If ${x_\mathrm{k}}^i$ and ${x_\mathrm{k}}^j$ are similar in the domain, then the corresponding W$_k(i,\ j)$ will be large. If the embeddings are separated in the new space, C will be large. Therefore minimizing C preserves the topology of the given domain

Combining all three cost functions together, we have the following equation as the cost function to be minimized.

$$(A + B)/C$$

Wang and Mahadevan [18] establishes proof of the theorem that embeddings to minimize this cost function are given by the eigenvectors corresponding to the d lowest eigenvalues of the generalized eigenvalue decomposition problem:

$$Z(\mu L + L_\mathrm{s})Z^T x = \lambda Z L_\mathrm{d} Z^T x$$

The d eigevectors form a mapping function matrix of dimenasion (p_1+p_2) x d. Thus the upper part of the matrix is the linking function f_1 to map SoMe domain to the latent space and the lower part of the matrix is the linking function f_2 to map ODHMC domain to the same latent space.

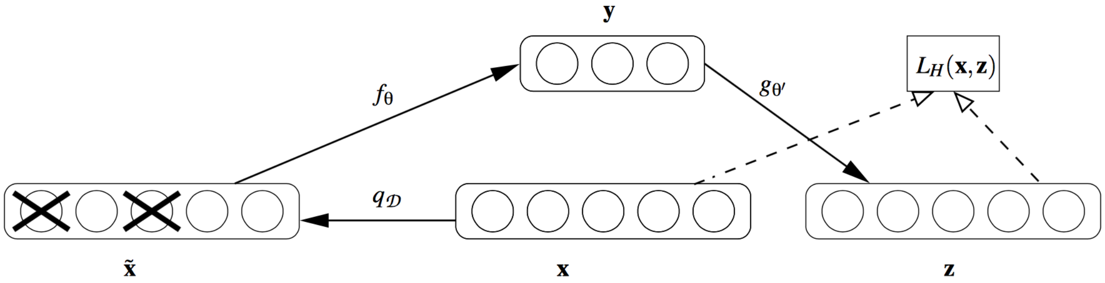


Figure 4.3: Illustration of a denoising autoencoder. (Copyright: Vincent et al. (2008))

4.2.3 Stacked Denoising Autoencoders

We use a deep-learning architecture, stacked denoising autoencoders, to explore its ability in generating the distributed data representation out of transformed features.

An autoencoder is comprised of an encoder function f_θ and a decoder function g_θ. The reconstruction of x is given by *r(x)* = g_θ*(*f_θ*(x))*, and autoencoders are typically trained to minimize certain reconstruction error, *loss(x, r(x))*. The alternative version is denosing autoencoders. This notion is first proposed by Vincent et al. (2008)[38]. In a denoising autoencoder (DAE), the input x will be stochastically corrupted into a vector $\hat{x}$, and the model is trained to denoise the error *loss(x, r(*$\hat{x}$*))*. The architecture of the denoising autoencoder is demonstrated in the Figure 4.3.

Once the denoising autoencoder has been built, another denoising autoencoder can be stacked on top of it, which takes the encoded output of the previous autoencoder as an input and repeats the training process. Once a stack has been trained, its parameters represent multiple levels of representations of x and other training or learning can be done on top of it. The detailed architecture is shown in the Figure 4.4. After training a first level denoising autoencoder, its learnt encoding function f_θ is used on clean input (left). The resulting representation is used to train a second level denoising autoencoder (middle) to learn a second level encoding function ${f_\theta}^2$. Then, the procedure can be repeated (right) to form the stack.

The reason for using denoising autoencoder is that when the number of hidden

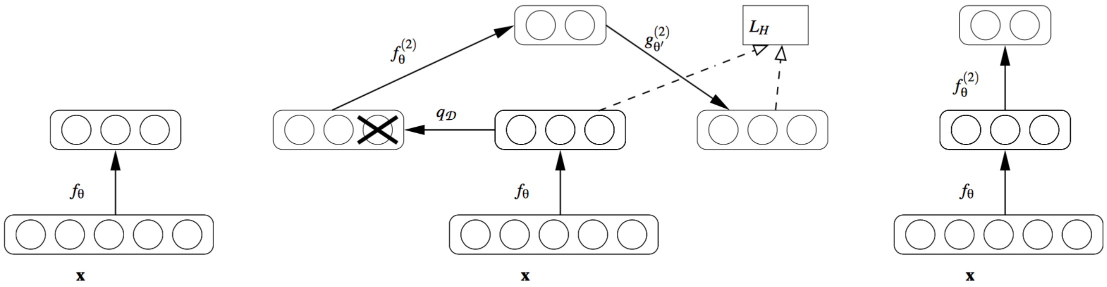


Figure 4.4: Illustration of a stacked denoising autoencoder. (Copyright: Vincent et al. (2008))

units is larger than the dimension of original input, the architecture will learn an identity matrix, a null solution. Also, for denoising autoencoders, the mean reconstruction error can be used to choose model capacity and do early stopping.

In our study, we do hyper-parameter tuning with grid-search technique. We finally report the results on a stacked denoising autoencoder of 3 layers, generating the distributed data representation of the original dimension 50, the first expanded dimension 70 and the second expanded dimension 90. Then the for the output layer is the logistic regression model.

Chapter 5 Experiments and Results

5.1 Experiments

In this section, we are presenting our experiment methods and the analysis methodology we use to measure the performance of different models.

5.1.1 Settings

We use all 250 labeled samples from the mini MyPersonality dataset and 180 labeled samples from our Alexa conversation dataset to conduct the following experiments.

- ***Traditional ML methods on Conversation data alone.*** First without any domain adaptation technique or auxiliry help from social media data, we experiment with a list of traditional machine learning methods on extracted conversation features alone. These methods include linear and non-linear methods: Logistic Regression (LR), Support Vector Machine (SVM) and Gradient Boosting Tree.

- ***Traditional ML methods on transformed Conversation data alone.*** Transformed conversation data is obtained by applying heterogeneous domain adaptation technique with the social media data as the auxiliary task. We choose dimension $n = 50$ to get the transformed feature matrix. The methods we tune are the same as above.

- ***Experiments with Feature Augmentation.*** We get the augmented features of both social media domain and conversation domain with feature matrices obtained by heterogeneous domain adaptation. We train models with three feature matrices: Target-Only, Source-Only, and Target-Source-Combined. The model we use in these experiments is logistic regression.

- ***Experiments with sDAE.*** We tune the hyper-parameters of the sDAE architecture to train the network with three feature matrices: Target-Only, Source-Only, and Target-Source-Combined, and compare the performance with models trained with augmented features.

5.1.2 Evaluation Metrics

The goal of our classifiers is to get the classification results correct instead of finding "potential extrovert/open-minded users". In this case, we put emphasis on strength of models to predict personality trait polarity, instead of retrieving positive or negative users. Therefore metrics like F1 score and AUC score which is specifically targeted for identifying small number of positive or negative samples from the whole population won't be applicable in this problem setting. Hence, we use tailored accuracy and precision rates to measure the performance, the strategy first proposed by Wei et al.[16].

- Model Accuracy Rate (MAR):

$$\mathbf{MAR} = \frac{number\ of\ correct\ predictions}{number\ of\ samples}$$

- Model Precision Rate of Positive Case (MPR@P):

$$\mathbf{MPR@P} = \frac{number\ of\ correct\ positive\ predictions}{number\ of\ positive\ samples}$$

- Model Precision Rate of Negative Case (MPR@N):

$$\mathbf{MPR@N} = \frac{number\ of\ correct\ negative\ predictions}{number\ of\ negative\ samples}$$

5.2 Results

In this section, we are going to go over all results we obtained from experiments and explain findings we have with analysis. We will also discuss the comparison between different approaches so studies after could dig further with one or several of methods explored in this research. **All results are obtained through 10-fold cross validation**.

5.2.1 Overall Results

A. Results on Original Conversation Features

We tune three traditional machine learning models implemented in scikit-learn [39] on original conversation features: logistic regression ($C = 1,\ C = 5,\ C = 10$), where C represents the strength of L2 regularization, support vector machine ($C = 10,\ C = 50,\ C = 100$), where C sets the value of margin in the hyperspace, and gradient boosting trees ($\alpha = 0.1,\ \alpha = 0.5,\ \alpha = 1$), where α represents the learning rate. Other parameters are set to be default. For prediction of both traits , it turns out that both LR and GBT could have similar satisfying performance while SVM obviously underperforms in this task, as shown in the following table. We report the best performance result for each model.

Traits	**Openness**			**Extraversion**		
Params	MAR	MPR@P	MPR@N	MAR	MPR@P	MPR@N
Logistic Regression						
$C = 10$	0.756	0.792	0.720	0.816	0.840	0.799
Support Vector Machine						
$C = 50$	0.637	0.678	0.596	0.702	0.736	0.673
Gradient Boosting Trees						
$\alpha = 0.5$	0.721	0.771	0.664	0.798	0.817	0.777

Table 5.1: Optimal Model Results on Original Conversation Features

Based on the results, we claim that both traits can be predicted with fair accuracy. On the original feature space, in terms of three metrics, extraversion seems to be more accurately predicted.

We can conclude that the logistic regression model with $C = 10$ L2 regularization has the most impressive performance on trait predictions, and extraversion trait is more easily predicted. For openness prediction, the best MAR score is 0.756, the best MPR@P score is 0.792, and the best MPR@N score is 0.720. For extraversion prediction, the best MAR score, the best MPR@P score and the best MPR@N score are 0.816, 0.840 and 0.814. We will use the results of the optimal logistic regression model as the baseline, and any statistically significant improvements (2-tailed t-test with 90% confidence level) in the following experiments will be marked with "*".

We also analyze the classifier-specific feature importance. The top 10 important features for openness and extraversion prediction tasks are listed with merit scores below.

Openness		**Extraversion**	
Feature Name	Average Merit Score	Feature Name	Average Merit Score
opn_pos_count	0.18 ± 0.007	wc_std	0.25 ± 0.007
opn_neg_count	0.152 ± 0.012	wc_avg	0.234 ± 0.012
suggested_topic_cov	0.122 ± 0.009	op_1	0.205 ± 0.016
no_ratio	0.121 ± 0.011	op_0	0.205 ± 0.016
states_cov	0.122 ± 0.009	ex_1	0.205 ± 0.016
avg_senti_score	0.077 ± 0.013	ex_0	0.205 ± 0.016
num_of_turns	0.059 ± 0.014	ext_pos_score	0.178 ± 0.01
senti_neg_score	0.041 ± 0.009	ext_neg_score	0.18 ± 0.009
opn_score_1	0.044 ± 0.016	senti_pos_count	0.101 ± 0.012
std_senti_score	0.036 ± 0.012	senti_neg_count	0.102 ± 0.009

Table 5.2: Feature Importance for Openness and Extraversion with Average Merit Score

From the table we can conclude the Pearson Correlation features (opn_pos_count, opn_neg_count, ext_pos_score, ext_neg_score) are crucial to classifying the polarity of users' personality traits. Some top openness positively correlated words include *love* (0.24), *travel* (0.18), and *avenger (the movie)* (0.17) . Some top extraversion positively correlated words include *call* (0.22), *want* (0.26), *favorite* (0.25).

B. Results on Transformed Conversation Features

We use manifold alignment heterogeneous domain adaptation mentioned in Section 4.2 to transform conversation data features with MyPersonality feature matrix, so original features are mapped into a latent subspace of dimension 50. We implemented the same set of experiments on this transformed feature matrix.

Traits	**Openness**			**Extraversion**		
Params	MAR	MPR@P	MPR@N	MAR	MPR@P	MPR@N
Logistic Regression						
$C = 1$	0.730	0.786	0.671	0.771	0.759	**0.784**
$C = 5$	**0.771**	0.798	**0.738**	0.761	0.748	0.781
$C = 10$	0.758	**0.814**	0.689	**0.777**	**0.819**	0.729
Support Vector Machine						
$C = 10$	0.647	0.681	0.609	0.608	0.670	0.544
$C = 50$	0.651	0.677	0.635	0.619	0.654	0.583
$C = 100$	0.658	0.664	0.649	0.616	0.666	0.583
Gradient Boosting Trees						
$\alpha = 0.1$	0.640	0.745	0.514	0.739	0.742	0.733
$\alpha = 0.5$	0.662	0.757	0.553	0.741	0.735	0.750
$\alpha = 1.0$	0.657	0.694	0.605	0.707	0.711	0.710

Table 5.3: Model Results on Transformed Conversation Features

As a result, logistic regression model still noticeably outperforms the other two. One noteworthy observation is that though performance of logistic models improve on openness prediction, the overall model performance on extraversion prediction drops. The value of optimal parameter decreases too, which could be the result of reduction of feature dimension.

C. Results on Feature Augmentation

As transformed social media features and transformed conversation features are now on the same subspace, we can do direct feature augmentation on both domains. We examine the influence of training on source domain on model performance on the target domain by training models separately on SRC-ONLY data, on TGT-ONLY data and SRC-TGT data, and comparing their performance on test target data. The prototype model we use for this experiment is logistic regression with L2 regularization of strength $C = 10$. Apparently, openness prediction task benefits more from training by combining both domains. All best results are bolded.

Traits	**Openness**			**Extraversion**		
Training Data	MAR	MPR@P	MPR@N	MAR	MPR@P	MPR@N
Baseline	0.756	0.792	0.720	0.816	0.840	0.799
SRC-ONLY	0.620	0.824	0.366	0.517	0.470	0.569
TGT-ONLY	0.721	0.744	**0.706**	**0.744**	**0.793**	0.702
SRC-TGT	**0.734**	**0.833***	0.621	0.737	0.685	**0.796**

Table 5.4: Model Results of Using Different Training Data: Simple Feature Augmentation

The MPR@P on openness trait is statistically significant compared to the baseline data.

D. Results on Stacked Denosing Autoencoders In the experiments of sDAE we use a masking noise and choose mean squared error as the loss function. The result is shown in the following table.

Traits	**Openness**			**Extraversion**		
Training Data	MAR	MPR@P	MPR@N	MAR	MPR@P	MPR@N
Baseline	0.756	0.792	0.720	0.816	0.840	0.799
SRC-ONLY	0.592	**0.887***	0.246	0.489	0.405	0.581
TGT-ONLY	**0.699**	0.754	**0.636**	**0.755**	**0.727**	0.793
SRC-TGT	0.650	0.793	0.471	0.726	0.642	**0.839***

Table 5.5: Model Results of Using Different Training Data: Stacked Denoising Autoencoders

The MPR@P score, when training only on source, is statistically significant compared to the baseline, the resulf of unbalanced class distribution in the SoMe domain. One impressive result is that the MPR@N socre, when training on source-target combination data, is statistically significant compared to the baseline.

In summary, it's practical to do binary polarity prediction on personality traits from open-domain human-machine conversations. In terms of 3 metrics under supervised setting, among linear and non-linear models we investigate, logistic regression with L2 regularization has the best performance no matter on original features or transformed features. On the original feature set, for openness prediction, topic-related hand-crafted features play the specific significant roles. For extraversion, wc features are the most important. For both dimension, several sentiment features and word Pearson Correlation features are crucial. These conclusions are consistent with our assumption that linguistic cues features' ability is restricted under the open-domain conversation settings. For domain adaptation, there are good techniques to learn data representation in a compressed space. However, deep-learning architectures aimed for unsupervised settings could have comparatively weak performance in the completely supervised setting with a few data.

5.2.2 Analysis of Transfer Learning (Domain Adaptation) in Personality Prediction

In the experiments we've tried both non-deep-learning and deep-learning approaches. Transfer learning is especially useful for task where there is only limited and biased data as it helps to choose a more generalized model. In terms of personality prediction task, transfer learning seems to be especially fit for traits less behavior-based, like "Openness": we consider people with high openness score not because of the way they talk (lots of questions, tend to talk more passionately, etc.) but because of what they talk (aesthetic subjects, art, music, tendency to accept new concepts and topics, etc). Therefore, with the social domain data as the helper, models tend to capture general latent contents better than specific format information of original data (like longer conversations, or more words per sentence).

Another potential advantage of domain adaptation which hasn't been explored in this study is its powerful ability to make use of unlabeled data. In the paper of heterogeneous domain adaptation with manifold alignment[18], the authors emphasize the usefulness of unlabeled data in the similarity and dissimilarity matrices calculation. Deep learning architectures like stacked autoencoders can even make better use of unlabeled data: the unlabeled samples could be used for unsupervised pre-training for each layer of this architecture, to minimize the reconstruction loss, and in the second stage the network could go through supervised fine-tuning. The first stage of unsupervised learning with tons of unlabeled samples have been proven to be extremely powerful on building up a robust stacked autoencoders architecture[40]. As it requires lots of human resources to do personality tests or label the data manually, deep learning architecture is a potential structure to be applied widely in a large open-domain human-machine conversation system.

5.2.3 Comparison between Feature Augmentation and sDAE

Feature augmentation is a very simple technique, just as the paper's title suggests "frustratingly easy". It requires 10 lines of Perl as claimed by the author[17], efficient and easy to interpret. One clear limitation of this algorithm is that it only applies to supervised setting, a practically rare case. Our study is completely under supervised setting by far, but if we'd extend the project further under semi-supervised or unsupervised settings, this algorithm will lose its significance. One claim we don't explore in this work is its ability to capture the clear differences in various domains as we only use social media data as the auxiliary task. This approach might be more powerful if there are more labeled data from other domains, like VLOG (video blog) domain, public speech domain, or task-oriented chat domain.

Comparatively, sDAE requires heavy training, whose time depends on the dimension of the network's reconstruction layers and the number of layers and units. Under supervised setting with only a few of clean, labeled data it's clearly not an ideal approach as its overall performance would be harmed by the first-stage reconstruction phase, but it's a more widely applicable approach if researchers also use vast, noisy, unannotated data because of its outstanding ability to do feature selection and learn a compressed, distributed data representation.

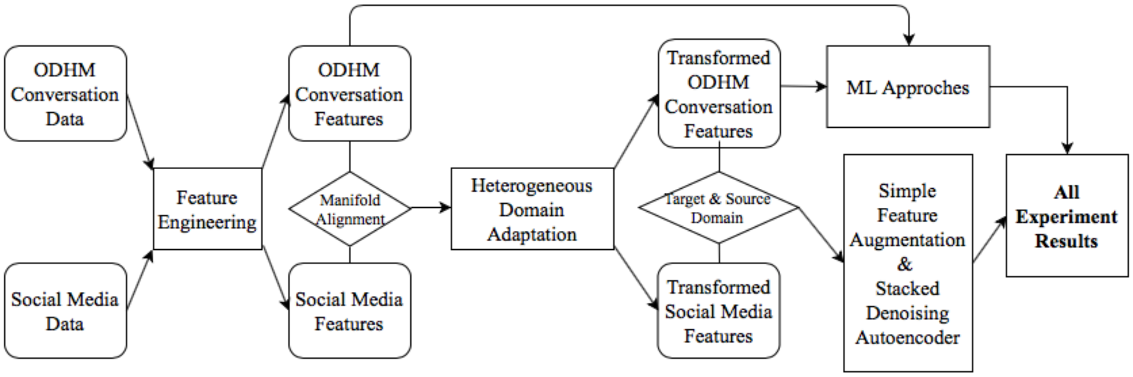

ODHM Conversation Data
Social Media Data
Feature Engineering
ODHM Conversation Features
Manifold Alignment
Social Media Features
Heterogeneous Domain Adaptation
Transformed ODHM Conversation Features
Target & Source Domain
Transformed Social Media Features
ML Approches
Simple Feature Augmentation & Stacked Denoising Autoencoder
All Experiment Results

Figure 5.1: Overview of the Model Development

Chapter 6 Conclusion and Future Work

In this study we explore the practicability to do personality predictions in the open-domain human-machine conversation domain. We also examine the significance of incorporating previous accomplished works on social media personality prediction as the auxiliary task to generate latent data representation (Heterogeneous Domain Adaptation, Section 4.2), augment current feature space (FEDA, Section 4.3) and construct compressed data representation (sDAE, Section 4.4). In other words, social media domain data is used as both assistant to transform feature space and to introduce inductive bias to generalize the final model.

From the experiment results, we conclude that it is feasible to prediction users' personality traits on conversations through a heterogeneous set of features: on our annotated dataset, our model achieves an accuracy score (MAR) of **0.756** in predicting Openness, a positive precision rate (i.e. identifying users with positive scores, MPR@P) of **0.792** and a negative precision rate (i.e. identifying users with negative scores, MPR@N) of **0.720**; in extraversion dimension it has even more impressive performance: the accuracy rate (MAR) is as high as **0.816**, the positive precision (MPR@P) **0.840** and the negative precision rate (MPR@N) **0.814**. In addition, we also conduct the same set of experiments on transformed feature set, the model performance in openness dimension is improved a little by 1.9% in MAR, 2.7% in MPR@P and 2.5% in MPR@N. In the extraversion dimension the performance is dropped in terms of these three criteria. Our explanation for this observation is that the generalized model could better capture the latent representation of content instead of format of data, concluded from the different ways these two traits are judged.

We choose one non-deep-learning and one deep-learning multi-task learning approaches to explore the benefits of using auxiliary tasks. For the FEDA algorithm, the openness prediction model trained on the combination of target and source domain has slightly better performance in terms of MAR score (0.721 vs. 0.734) and MPR@P score (0.744 vs.0.833); the extroversion prediction model trained on the combination is able to capture the negative examples better. For sDAE, it doesn't have very impressive performance as previous methods. This approach is not ideal in this case, as our study is conducted in a completely supervised setting, the power of unsupervised learning of autoencoders is eliminated, and we don't have unlabelled data for the autoencoders to do feature selection at the first stage either.

Our research acts as a beginning for this field as it's the first project to do personality prediction in open-domain human-machine conversation settings. The task itself is crucial with the fast development of intelligent assistants, to be part of a user profiling system to provide highly customized personal service.

There are several open questions related to this research. First, while our model has promising performance in predicting extroversion and openness, we have not experiments with the other three personality traits (consciousness, agreeableness and neuroticism). How to predict those still remains an open question. Second, our study is carried out in a completely supervised setting. During the Alexa Prize contest we have collected orders of magnitude more conversations, but the value of these unannotated data has not yet been explored. Third, we develop independent prediction models on each trait but haven't investigated the effects of co-prediction as there might be covariance among personality traits.

In summary, my thesis has the following contributions:

- (a): This research confirms the practicability to predict personality traits polarities from open-domain human-machine conversations, and presents experiment results with both linear and non-linear models.

- (b): I design and compile the list of useful features to represent conversation data, and prove the feature importance in the experiment.

- (c): I incorporate multiple domain adaptation techniques to transfer the ODHMC domain onto a latent space with the external domain, SoMe domain, as the auxiliary task to perform feature transformation and feature augmentation, and develop models on new common feature space to capture signals from both domains.

Bibliography

1. Zhang, X. & Zhao, H. Cold-Start Recommendation Based on Speech Personality Traits. *Journal of Computational and Theoretical Nanoscience* **14,** 1314–1323 (03/2017).
2. J. Gill, A., Nowson, S. & Oberlander, J. *What Are They Blogging About? Personality, Topic and Motivation in Blogs* in *Proceedings of the The International AAAI Conference on Weblogs and Social Media* (2009), 18–25.
3. Flekova, L. & Gurevych, I. *Personality profiling of fictional characters using sense-level links between lexical resources* in *Proceedings of Empirical Methods in Natural Language Processing* (2015).
4. Arnoux, P. *et al. 25 tweets to know you: A new model to predict personality with social media* in *In Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM* (2017).
5. Camacho, L. & Alves-Souza, S. *Social network data to alleviate cold-start in recommender system:A systematic review* 07/2018.
6. Gao, R., Hao, B., Bai, S., Li, L. & Li A.and Zhu, T. *Improving user profile with personality traits pre- dicted from social media content* ACM, 2013.
7. Higashinaka, R. *et al. Towards an open-domain con- versational system fully based on natural language processing* in *COLING* (2014), 928–939.
8. Danescu-Niculescu-Mizil, C. & Lee, L. *Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs.* in *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011* (2011).
9. Budzianowski, P. *et al. MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling* in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2018).
10. Logacheva, V. *et al.* A Dataset of Topic-Oriented Human-to- Chatbot Dialogues.
11. myPersonality (07/2012).
12. Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A. & Booth, R. J. The development and psychometric properties of LIWC2007. (2007).
13. Schwartz, H. A. *et al.* Personality, gender, and age in the language of social media: The open vocabulary approach. *PLOS ONE* (2013).
14. Skowron, M., Tkalčič, M., Ferwerda, B. & Schedl, M. *Fusing social media cues: Personality prediction from Twitter and Instagram.* in *Proceedings of the 25th interna- tional conference companion on World Wide Web.* (2016), 107–108.

15. Arnoux, P. *et al. 25 tweets to know you: A new model to predict personality with social media* in *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM* (2017).

16. Wei, H. *et al.* Beyond the words: Predicting user personality from heterogeneous information. 305–314 (2017).

17. Daume III, H. *Frustratingly easy domain adaptation.* in *ACL* (2007).

18. Wang, C. & Mahadevan, S. *Heterogeneous domain adaptation using manifold alignment.* in (2011).

19. Norman, W. Toward an adequate taxonomy of personality attributes: Replicated factor structure in peer nomination personality rating. *Journal of Abnormal and Social Psychology,* 574–583 (1963).

20. Allport, G. W. & Odbert, H. S. Trait names: a psycho-lexical study. 171–220 (1936).

21. Hogan, R., Curphy, G. J. & Hogan, J. what we know about leadership: Effectiveness and personality. *American Psychologist, 49(6),* 493–504 (1994).

22. Komarraju, M. & Karau, S. J. The relationship between the Big Five personality traits and academic motivation. *Personality and Individual Differences,* 557–567 (2005).

23. Marshall, T. C., Lefringhausen, K. & Ferenczi, N. The big five, self-esteem, and narcissism as predictors of the topics people write about in Facebook status updates. **85,** 35–40 (2015).

24. Oberlander, J. & Nowson, S. *Whose thumb is it anyway? classifying author personality from weblog text.* in *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics* (2006).

25. Pennebaker, J. W. & King, L. A. Linguistic styles: Language use as an individual difference. *Journal of Personality and Social Psychology* **77,** 1296–1312 (1999).

26. Mairesse, F., Walker, M., Mehl, M. & Moore, R. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research 30,* 457–500 (2007).

27. Luyckx, K. & Daelemans, W. Personae: a corpus for author and personality prediction from text. (2008).

28. Noecker, J., Ryan, M. & Juola, P. *Psychological profiling through textual analysis.* in *Literary and Linguistic Computing.* (2013).

29. Golbeck, J., Robles, C., Edmondson, M. & Turner, K. *Predicting personality from twitter.* in *Proc of the 3rd IEEE Int Conf on Soc Comput.* (2011), 149–156.

30. Kim, Y. *Convolutional neural networks for sentence classification.* in *In Proceedings of the 2014 Confer- ence on Empirical Methods in Natural Language Processing (EMNLP)* (2014), 1746–1751.

31. Mehl, M., Pennebaker, J., Crow, M., Dabbs, J. & Price, J. he Electronically Activated Recorder (EAR): A device for sampling naturalistic daily activities and conversations. **33,** 517–523.

32. Fang, H. *et al. Sounding Board – University of Washington's Alexa Prize Submission.* in *Alexa Prize Proceedings* (2017).

33. Tigunova, A., Yates, A., Mirza, P. & Weikum, G. *Listening between the Lines: Learning Personal Attributes from Conversations* in *Proceedings of WWW2019* (2019).

34. TextBlob: Simplified Text Processing. (06/2017).

35. Pennington, J., Socher, R. & Manning, C. D. GloVe: Global Vectors for Word Representation.

36. Fast, E., Chen, B. & Bernstein, M. S. Empath: Understanding Topic Signals in Large-Scale Text. (2016).

37. Le, Q. & Mikolov, T. *Distributed rep- resentations of sentences and documents.* in *Pro- ceedings of the 31st International Conference on Machine Learning (ICML 2014)* (2014), 1188–1196.

38. Vincent, P., Larochelle, H., Bengio, Y. & Manzagol, P. *Extracting and Composing Robust Features with Denoising Autoencoders* in *Proceedings of the Twenty-fifth International Conference on Machine Learning* (2008), 1096–1103.

39. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12,* 2825–2830 (2011).

40. Bengio, Y., Lamblin, P., Popovici, D. & Larochelle, H. *Greedy Layer-Wise Training of Deep Networks* in *Advances in Neural Information Processing Systems 19 (NIPS'06)* (2006), 153–160.