Predicting Sales Email Responders using a Natural Language Model

Markus Krause, Anand Kulkarni
UC Berkeley, ICSI; LeadGenius
public.markus.krause@gmail.com, anand@leadgenius.com

Abstract

Email is a standard and popular means of establishing potential business relationships between salespeople and future customers, but it is difficult for machines to generate messages that are as convincing as a human author. We take first steps towards the automatic generation of human-quality emails by presenting a model predicting if an email will be successful in eliciting a response. In this paper we propose a natural language model that predicts whether a human-authored sales e-mail will get a response from a previously uncontacted recipient. We test our algorithm with a set of 116 outbound sales e-mails used in practice. Our algorithm is successfully able to predict if an e-mail in this set received a response with an F1 score of 0.81. This work provides initial steps in understanding how to automate convincing communication over email between humans and computers.

Introduction

Email is a standard means of communication all over the world, and an increasingly popular means of communication in sales and marketing. Salespeople frequently "cold email" potential clients or partners attempting to open a new relationship, in place of using a telephone "cold call". While salespeople's success depends on whether their communication elicits positive responses, this domain is still largely a black art. Little is known from a computational perspective on what types of communication will elicit a response from a human recipient. If we are able to construct a model that predicts whether a recipient will reply with interest to an email, we can better target emails and reduce the volume of email sent unnecessarily. More fundamentally, this question is important not only from a sales perspective but also in providing a computational basis for understanding communication over email.

We set up a natural language model and trained the model on a sample set of 116 outbound sales emails used in a commercial sales setting, including 43 which received a positive reply and 76 which received no response. The emails varied in length, subject line, recipient profile and industry, but all had the common attribute of attempting to open a new conversation with a potential prospective business partner for the first time and establish a new business relationship.

We generated one feature vector per mail and sample these vectors into train and test sets using stratified random sampling (Muench, 1954). We generate 10K train/test sets to estimate means and standard deviations for F1 and Kappa scores. Test sets contained 25% of all samples while training sets contained the remaining 75%. Finally, we estimate the performance of our model using F1 and Cohens' Kappa scores and report mean and SD for both scores as well as the normalized confusion matrix.

We used a random forest classifier, because they directly handle multiple classes, and are less sensitive to outliers (Breiman, 2001). In our evaluation, we only used a fixed numbers of training samples per user. Our classifier generated 500 random trees per forest using Gini impurity (Breiman, 1996) as split criterion.

Language Model

We base our linguistic model on a feature set that has previously been used to investigate writing styles in educational settings (Kilian, Krause, Runge, & Smeddinck, 2012; Krause, 2014). We use the following set of features: length frequencies (word length, sentence length), emotional content (valence and arousal), language specificity frequency, part of speech frequency, and sentence mood. We preprocessed all reviews with the NLTK part-of-speech (POS) tagger (Bird, Klein, & Loper, 2009). We then filtered stop words and words not in Wordnet (Miller, 1995). Wordnet is a natural language tool that provides linguistic information on more than 170,000 words in the English language. We also lemmatized the remaining words to account for different inflections.

Part of Speech Tag Frequency: For this feature set we use the Penn Treebank part of speech tag set. We use *pattern.en* to extract these tags. We calculate the relative frequency of each tag. Giving a total of 35 features.

Text length: the first two feature sets we use the frequency of number of letters in words and the frequency of number of words per sentence. For word length frequency we considered only those words that have a *Wordnet* entry and are not stop words. Furthermore we group words longer than 20 characters in one group so word length frequency gives us 20 features. The sentence length was measured including all words returned by the POS-tagger. We grouped sentences longer than 30 words into one group, so sentence length frequency gives us 30 individual features.

Emotionality: The next two feature sets we looked at were valence and arousal. Valence refers to whether a text is positive, negative, or neutral, and arousal represents how strong the valence is. We use 5 levels of arousal and valence both ranging from 1 to 5 so 10 features total. We used *pattern.en*, a tool based on *NLTK*, to extract valence and arousal.

Specificity: Another feature set we explored was specificity, which refers to how specific the words in a text are. We measured specificity by determining how deep each word appears in the *Wordnet* structure. Words that are closer to the root are more general (e.g. *dog*) and words deeper in the *Wordnet* structure are more specific (e.g.). Word depth ranges from 1 to 20 (20=most specific).

Sentence Mood: the last features we considered involves looking at moods of sentences. Each sentence was classified as either indicative (written as if stating a fact), imperative (expressing a command or suggestion), or subjunctive (exploring hypothetical situations). We again used *pattern.en* to extract sentence mood.

Results

In our experiment we reached a substantial Kappa score of 0.68 (SD = 0.06) which translates to an F1 score of 0.81 (SD = 0.07). As the confusion matrix in Figure 1 shows the error rate leans more towards false positive detecting a bounce. As the training data was normalized to an equal class distribution the reason must be found in the data itself. As this is a preliminary study we do not yet have an explanation which features are causing this effect. As we are not aware of any comparable experiments we did not compare our results to other approaches.

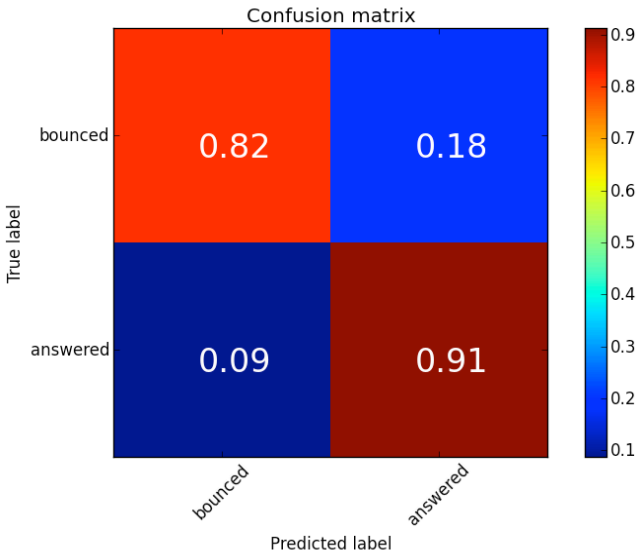

Figure 1: Confusion matrix of the experiment. The error rates lean more towards false positive detecting a bounce. The Kappa score for the matrix is 0.68 and an the F1 score is 0.81.

References

Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python.

Breiman, L. (1996). Technical note: Some properties of splitting criteria. Machine Learning, 24(1), 41–47.

Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5–32.

Kilian, N., Krause, M., Runge, N., & Smeddinck, J. (2012). Predicting Crowd-based Translation Quality with Language-independent Feature Vectors. In HComp'12 Proceedings of the AAAI Workshop on Human Computation (pp. 114–115). Toronto, ON, Canada: AAAI

Krause, M. (2014). A behavioral biometrics based authentication method for MOOC's that is robust against imitation attempts. In Proceedings of the first ACM conference on Learning @ scale conference - L@S '14 (pp. 201–202). Atlanta, GA, USA: ACM Press.

Miller, G. A. (1995). WordNet: a lexical database for English. Communications of the ACM, 38(11), 39–41.

Muench, H. (1954). Sample Survey Methods and Theory. American Journal of Public Health and the Nations Health, 44(5), 687–688.