Sentiment Analysis and Visualization of Free-Response Survey Data

Joshua Posey[*]
Morehouse College

Ronald Metoyer[†]
Oregon State University

Abstract

In this proposal, we describe in-progress work on a project in which we have designed and implemented a web application for users to interactively view and explore survey data. This proposal focusses on the display of responses to open-ended questions which we call "free-response" questions. In order to visually display the free response data, we have developed a sentiment analysis pipeline to categorize the polarity of responses (e.g., are they positive or negative in nature), and we have designed visualization methods to display the data while maintaining privacy. We present the pipeline and plans for two visual representations designed to compute and communicate the sentiment of free-response survey data respectively.

1 Introduction

Survey data is being collected at an incredible rate given the rise of tools such as SurveyMonkey. Many surveys include free-response questions in which respondents provide free-form written answers to open-ended questions. While these data are often rich and may shed light on other survey questions (e.g., accompanying likert-scale responses) they are typically not shown in published reports due to the qualitative nature of the data and to the need to maintain the privacy of the respondents.

In this project, we have collaborated with the Computing Research Association's (CRA) Center for Evaluating the Research Pipeline (CERP) [1] to design and implement an interactive web site to allow the public to explore the results of their surveys. The surveys generally consist of a set of likert-scale questions and free-response questions.

We propose a method to convert free-response data to numerical values that can be visualized without violating privacy. The free-response data are converted to numerical values using a sentiment analysis approach that assigns a polarity (positive or negative strength) to each word of the response and ultimately to the entire response through aggregation. In the following sections, we describe the sentiment analysis approach and present our plans for visualizing the free response data.

2 Sentiment Analysis

Our sentiment analysis relies on the lexical tool, SentiWordNet, that computes sentiment scores for words in the English language [3]. Sentiment analysis using SentiWordNet requires three primary steps. First, the words of the free-response text must be tagged to identify the part of speech for each word. Second, each word is run through algorithms to identify the word stem, make it singular, and properly treat words used in the negative sense. Finally, the word is looked up in the SentiWord database to obtain its polarity as well the magnitude of this polarity (see Figure 1). In the following sections, we will describe each step. To ground the discussion, consider following fictitious survey question response that we will refer to in the following sections: *"I struggled to understand the teaching assistant. He did not provide clear explanations for the solutions."*

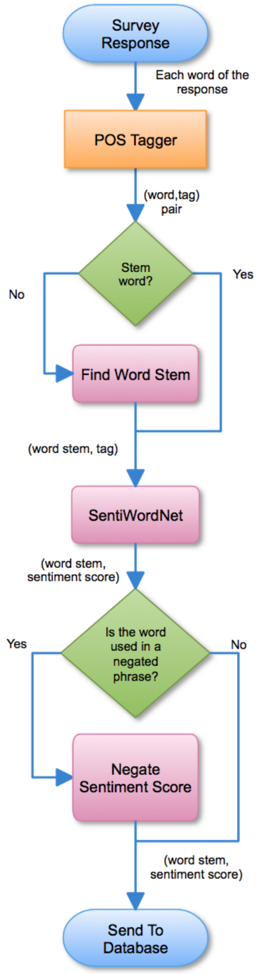

Figure 1: Flowchart for the sentiment analysis pipeline.

2.1 Part of Speech Tagging

SentiWordNet assigns a sentiment value to a word based on how that word is used in the sentence (i.e., what part of speech it belongs to). We first break the response into words and identify the part of speech for each word using the LBJPOS tagger from the University of Illinois [6],which produces tags that are appropriate to the context in which the word was used. For example, the word 'good' may be tagged as a noun or an adjective based on how it is used in the sentence. For each free response, the POS tagger produces a list of pairs $< word, tag >$ where there are 36 possible tags.

The next step of the process involves converting the tags into a format that is compatible with SentiWordNet. SentiWordNet han-

[*] e-mail: posey@engr.oregonstate.edu
[†] e-mail: metoyer@eecs.oregonstate.edu

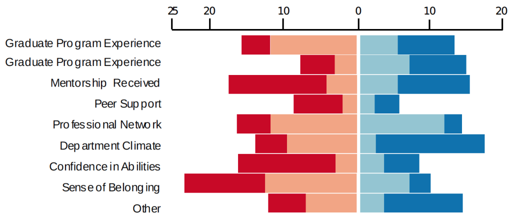

Figure 2: Example of a stacked-likert scale visualization. Each row corresponds to a likert-scale question. The red bars on the left represent the percentage of responses of values 1 and 2 (dark and light red respectively) and the blue bars on the right represent the percentage of responses of values 4 and 5 (light blue and dark blue respectively). In a polarized likert scale question, the left side might represent 'negative' responses while the right side might represent 'positive' responses.

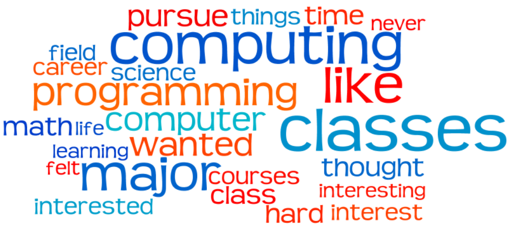

Figure 3: Tag cloud example constructed from synthetic free response data. In this example, the top 25 most frequent words are shown and each word has been encoded with a color that matches its aggregate polarity for this particular question (blue shades for positive and red shades for negative) and a size that represents its frequency. This tag cloud was constructed using wordle.net [5]

.

dles a limited set of tags; adjectives (a), nouns (n), verbs (v), adverbs (r) and adjective satellites (s), each corresponding to a subset of tags produced by LBJPOS. We map each LBJPOS tag to the appropriate umbrella tag in SentiWordNet.

2.2 Stemming

SentiWordNet typically contains only the stems of words in singular form in its database. Therefore, when necessary, we must perform operations to find the stems of the words used in the free-response text. In the example response above, the word "struggled" is the past tense form of the word "struggle" and "explanations" is a plural form of "explanation". In these cases, we use stemming algorithms [4, 2] that produce the stem words that are compatible with SentiWordNet.

2.3 Negations

SentiWordNet handles the sentiment analysis on a single word basis and does not account for negations. Negations imply the opposite sentiment of what is normally expected. In the example response used above, the sentence "He did not provide clear explanations for the solutions." would produce an aggregated positive sentiment if the word "not" isn't taken into account. To account for this, once a word is passed to SentiWordNet, it is checked against a list of negation words. There are two methods for handling negation words. The first method assumes that the response includes only one sentiment on a single topic, and thus negates the remainder of the words that follow a negation word. The second method negates a word only if the word immediately before it is a negation word. To negate a word's polarity, its score is simply multiplied by -1. We are currently experimenting with these methods as well as using a combination of them to handle negations.

2.4 Storing Sentiment Data

We store sentiment scores for each individual word as well as entire responses. To compute the sentiment score for an entire response, we simply sum the sentiment values of all words in the response. In addition, we keep track of the number of times each word is mentioned in response to a particular survey question. This frequency data will be used for visualization purposes as described in the following section.

3 Visualization

We can visualize the content of free response text in multiple ways. One option is to visualize the aggregate response for any question by visually indicating the percentage of negative sentiment responses as compared to the percentage of positive sentiment responses. This is similar to visualization of likert-scale responses that typically contain scales with positive and negative values. For example, on a 5-point likert scale, 1-2 may be positive, 3 neutral, and 4-5 negative (or vice versa). In our application, we represent polarized likert scale responses using a stacked-likert design (See Figure 2).

We can similarly represent aggregated free-response data. For any free response question, we bin the sentiment values for each response into 4 bins representing very negative, negative, positive and very positive. The aggregate number of responses in these bins can then be used to compute the percentage of responses in each bin and rendered along with the likert-scale response questions as shown in Figure 2.

The second option is to show the free response text directly. To avoid showing any single response in its entirety (to preserve anonymity), we aggregate all responses to a single free-response question by recording all words used in the responses and counting how many times they appear. This word frequency information can then be displayed using a tag cloud representation (See an example in Figure 3).

4 Conclusion and Future Work

We have presented a work-in-progress pipeline for visualizing free-response text data from surveys. Our approach maintains privacy and provides a measure of polarity for the free response text data. This polarity information can then be used to render aggregate responses in a stacked-likert scale like representation and actual response content in a tag cloud representation. The presented pipeline has been completed and is in a test phase. We are currently integrating the visualization components.

References

[1] Center for evaluating the research pipeline. http://www.cra.org/cerp. Accessed: 2014-08-08.

[2] Inflection. https://github.com/kevinweil/elephant-bird. Accessed: 2014-08-08.

[3] Sentiwordnet. http://sentiwordnet.isti.cnr.it. Accessed: 2014-08-06.

[4] Stemming. http://chianti.ucsd.edu/svn/csplugins/trunk/soc/layla/WordCloudPlugin/trunk/WordCloud/src/cytoscape/csplugins/wordcloud/Stemmer.java. Accessed: 2014-08-08.

[5] Wordle. http://www.wordle.net/. Accessed: 2014-08-08.

[6] D. Roth and D. Zelenko. Part of speech tagging using a network of linear separators. In *Coling-Acl, The 17th International Conference on Computational Linguistics*, pages 1136–1142, 1998.