11-2017

Intent Detection through Text Mining and Analysis

Samantha Akulick
Sheridan College, samantha.akulick@sheridancollege.ca

El Sayed Mahmoud
Sheridan College, elsayed.mahmoud@sheridancollege.ca

Intent Detection through Text Mining and Analysis

Samantha Akulick
Faculty of Applied Science & Technology
Sheridan College
Oakville, Canada
akulick@sheridanc.on.ca

El Sayed Mahmoud
Faculty of Applied Science & Technology
Sheridan College
Oakville, Canada
elsayed.mahmoud@sheridancollege.ca

***Abstract*— this work investigated using n-grams, parts-Of-Speech and Support Vector machines for detecting the customer intents in the user generated contents. The work demonstrated a system of categorization of customer intents that is concise and useful for business purposes. We examined possible sources of text posts to be analyzed using three text mining algorithms. We presented the three algorithms and the results of testing them in detecting different six intents. This work established that intent detection can be performed on text posts with approximately 61% accuracy.**

Keywords—intent detection; text mining; support vector machines; N-grams; parts of speech

I. INTRODUCTION

Root-cause analysis of user-generated content in social media helps to answer why customers dislike a product or service. However, identifying customer intent after the root-cause analysis is a much more valuable piece of information in terms of marketing and customer service [4]. This information allows companies to adapt their products and policies to customer intentions. The current data analytics product of Kaypok Inc. offers a variety of innovative solutions for root-cause analysis. Through this applied research project with Sheridan College, Kaypok aims to develop an add-on service that identifies customer intent via-à-vis its product in user-generated content. The remaining sections of this paper presents background of this project, three algorithms for detecting customer intents, results of testing the three algorithms, analysis of the results and conclusion

I. BACKGROUND

The concept of "context of situation" was first proposed by Malinowski and Gardiner in 1923 and used to understand utterances. Firth also drew attention to this theory as a central component of his linguistic approach [1]. He argued that, "Language represents a set of events in which speakers uttered an action one learned in doing things." [1]. Many wish-recognition [5] and intention-recognition approaches focus on the intractable form of the problem by assigning intention to a sequence of user behaviors [3]. However, we choose to focus on intention analysis in text mining at the granularity level of a single sentence.

Between 2012 and 2015, Kaypok Inc. developed a variety of tools and solutions for root-cause analysis using natural language processing and deep-learning techniques that automatically discover the "why" of any conversation in user-generated content in social media and the relationship with the customer. This technique can be used to answer why customers dislike a particular product or service.

The next step was to offer is the ability to detect customer intent. For instance, after Kaypok discovered why customers dislike a product or service (root-cause analysis), the product or service provider needs to know the customer's intent or action taken. These intentions include: Do they want to switch to another product? Do they use proactive language, indicating they are looking to switch to another product or service?

II. METHODS

A. Categorization

Carlos & Yalamanchi [2] found ten categories of intent useful to business: wish, purchase, inquire, compare, praise, criticize, complain, quit, direct, sell and other. Our research is looking purely for intent without subject or the underlying reason – with this understanding, some of these distinctions become unnecessary. For example: the fundamental difference between a complaint and a critique is the reason behind the argument; without the reason, they are both simply negative expressions. Six intents without subject were found, which fall into three categories:

- *Purchase / Quit*: The writer desires to either buy or sell a product, or sign up for / cancel a service.
- *Recommend / Warn*: The writer is actively recommending the product or service to others, or warning them against buying or pursuing it.
- *Praise / Complain*: The writer is taking no action but is giving an evaluation of the product or service.

After human analysis of the test data, it was determined that intent does not fit neatly into such categories, but is almost always a combination of two of the above – particularly given that most reviewers and social media posters do not explicitly state their intentions. Therefore, we determined the results of our algorithm should be the top two intents for a post.

B. Data Sources

Measurable expressions of intent exist in many online forums. Twitter, Facebook, Blogger and similar sites allow users to post about anything – including their experiences with and opinions of companies, products and services. There also exist more directed websites like ProductReview.com, whose primary purpose is to collect and host reviews of all types of product, or Oyster.com, which collects reviews of hotels exclusively.

The format of what users submit varies across platforms – some allowing for criteria specific ratings, some ratings from 0-10, some 0-5, and as with the undirected social media sites, there may be no ratings at all. The comment element between most websites is the written element – be it a short 140 character "tweet" from Twitter, a longer Facebook or blog post, or the written component of a directed review.

For testing purposes, Kaypok provided 358 review posts from banking institutions and 100 review posts from Amazon,

C. Dictionaries

While context is a key tool in human analysis, we assumed contextless analysis of such posts is possible based on the use of language. A number of negative adjectives or expletives generally suggest someone is extraordinarily unhappy with a service, while a number of positive adjectives generally suggest the opposite. These sentiments, combined with key verbs (quit, leave, recommend, join, buy, sell) form a reasonable representation of the intent in a post.

In order to perform the above type of analysis, dictionaries of different types of words were required. These dictionaries were created independently of the test data, using a combination of many online sources (dictionary.com, thesaurus.com, enchantedlearning.com, adjectivesstarting.com, myvocabulary.com) and the experience of a native English speaker. The dictionaries used in the algorithm are as follows:

Positive words (great, wonderful, amazing), negative words (suck, terrible, awful), quitting words (leave, quit, cancel), joining words (buy, join, signup), recommending words (suggest, recommend, propose), warning words (warn, avoid, condemn), and inverse words (not, isn't, can't).

When using n-gram-based algorithm, the dictionaries are not limited to single words, but contain n-grams of the same sentiment (good service, great service, excellent service.) For testing purposes, only monograms and bi-grams were used.

III. ALGORITHMS

Three algorithms were created based on different methods of text mining. The first algorithm analyzed the text using n-grams. The second algorithm analyzed the text using Part-Of-Speech (POS) tagging. The third algorithm analyzed the text by combining N-grams and Support Vector Machines.

A. Ngrams-based Algorithm

Algorithm 1 shows the detailed steps for identifying the two likely intents of a post based on the said intent within each sentence in the post. The algorithm splits the post into a list of sentences and analyses each sentence into a list of n-grams. The n-grams are checked against predefined dictionaries to identify the intents associated with these n-grams. Weight values are given to sentence and to each intent within each sentence. The algorithm determines the two likely intents of the post based on the highest weight values. This work investigated using two lengths of n-grams: unigrams and bigrams.

Algorithm NgramsIntentDetector (*p*)
Input: a text post (tweet/review/comment) *p*
Output: The two likely intents of the post *P*
Split the post *p* into a list of sentences *S*.
for *i*←0 **to** length of *S* **do**
  Break the sentence S[*i*] into a list of n-grams *NG*
  for j←0 **to** the length of the *NG* list **do**
    Check the n-gram *NG*[*j*] against dictionaries.
    if a match for *NG*[*j*] in the dictionary is found **then**
      Check that the *NG* [*j-1*] was not inverse
      (For example a match of 'buy' is inverse if the sentence reads "not buy" or "don't buy").
      if *NG* [*j-1*] in [not, don't] **then**
        Update the sentence record S[i] by incrementing the weight of the dictionary type inverse (recommend becomes warn, etc).
      else
        Update the sentence record S[i] by incrementing weight of the corresponding dictionary type.
      end if
    end if
    {increment the counter j}
  end for
  Determine the weight of said intent within the sentence S[i] by tallying all weights that contribute to said intent and subtracting those which counter it.
  {increment the counter *i***}**
end for
Weight each sentence s according to its position in the post (first and last sentences carry the most weight, the ones in the middle carry less. This is based on observation of the test data).
Determine the final intents of the post by tallying the weighted values of the sentences.
The two highest intent values are the two likely intents of the post P.
return the two likely intents of the post P

Algorithm 1: identifying the two likely intents of a post using N-grams analysis and a predefined dictionary.

Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors*).

B. Part-Of-Speech (POS) Algorithm

Algorithm 2 shows the detailed steps for identifying the two likely intents of a post using part-Of-Speech tagging. The algorithm uses "patterns," a unique combination of POS tags and dictionaries, to identify more variable expressions in the text. A phrase like "I really hated it" would match a pattern with the POS labelled as "B-NP, B-ADVP, B-VP, I-VP", compared against the corresponding dictionaries "none, not inverse, negative, none". The same pattern would match "He said awful things" and "She experienced terrible service."

The algorithm splits the post into sentences then tag the parts-of-speech for the sentence. The combinations of the parts-of-speech are checked against predefined dictionaries to identify the intents within the sentence. Weight values are given to sentence and to each intent within each sentence. The algorithm determines the two likely intents of the post based on the highest weight values.

Algorithm PosIntentDetector (*p*)
Input: a text post (tweet/review/comment) *p*
Output: The two likely intents of the post *P*
Split the post *p* into a list of sentences *S*.
for *i*←0 **to** length of *S* **do**
    Tag the Parts-Of-speech (POS) for the sentence S[*i*]
    Check the POS in the sentence S[i] with the predefined POS patterns.
    if the POS pattern is present **then**
        Iterate through the sentence S[i] to find all instances of the said pattern.
        Compare the words in said instance with the dictionaries provided
        if *a positive match found in the dictionaries* **then**
            Update the sentence record S[i] by incrementing weight of the corresponding dictionary type.
        end if
    end if
    Determine the weight of said intent within the sentence S[i] by tallying all weights that contribute to said intent and subtracting those which counter it.
    {increment the counter *i***}**
end for
Weight each sentence s according to its position in the post (first and last sentences carry the most weight, the ones in the middle carry less. This is based on observation of the test data).
Determine the final intents of the post by tallying the weighted values of the sentences.
The two highest intent values are the two likely intents of the post P.
return the two likely intents of the post P

Algorithm 2: identifying the two likely intents of a post using POS Tagging and predefined dictionaries.

C. Support Vector Machines

Fig 1 shows the steps for the third algorithm to identify the two likely intents of a post by combining Support vector machines with polynomial kernel and the Ngrams. The algorithm uses 70% of the posts for building a model (training phase) and 30% of the data for testing the resulting model. For building the model, the post sentences have split into a sequence of unigrams. Each sequence has been associated with an intent. All the sequences have been passed to the support vector machines for constructing a model that enable us to detect the intent. The constructed mode has been tested using 30% of the posts after splitting their sentences into unigrams.

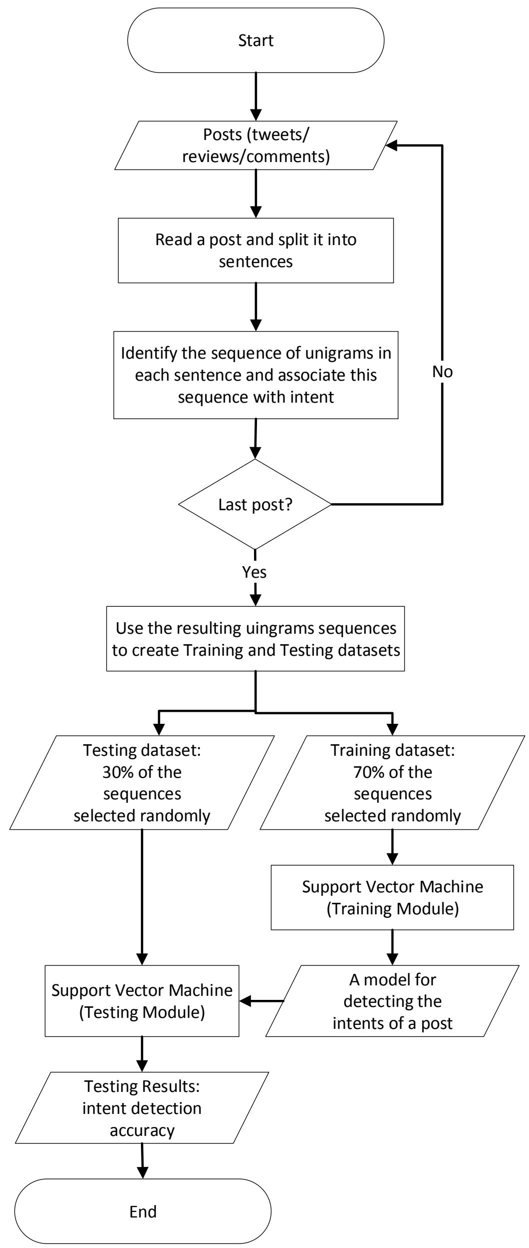

Fig 1: Steps of Algorithm 3 that uses Support Vector machines to detect the customer intents

IV. TESTING

In order to test the thee algorithm, a selection of 200 of the testing reviews were analyzed and labelled manually, indicating up to two intents for each post. Algorithms 1, 2 and 3 were run on the same selection of reviews. A match indicates one or both of the manual intents being identified, a partial match where the algorithm identifies one correct and one incorrect intent, unidentified where the algorithm failed to find any intent (but the human did) and a mismatch where all algorithmic results were incorrect.

V. RESULTS

Fig 2 shows the accuracy of using the three algorithms: Ngrams, Parts-Of-Speech, and Support vector Machines combined with n-grams to detect the customer intents in user generated content such as online posts. Using Ngrams achieved the highest accuracy compared to the parts of speech and SVM. A significant overlap among the patterns of the parts of speech has been noticed. This explains the low accuracy of detecting the intents based on parts of speech. We don't expect the poor performance of the SVM in this problem because of the SVM ability to maximize the margin when selecting the separators among the classes. However, we still have the motivation to investigate using different input data structures and SVM setting in future.

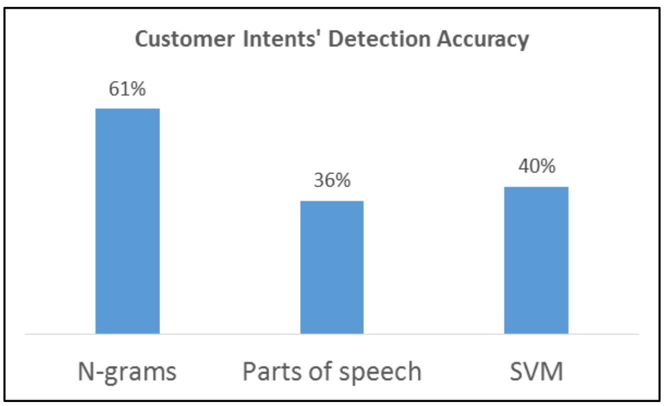

Fig 2: Customer intents Accuracy using Ngram, Parts-Of-Speech and Support vector Machines (SVM)

Also, a substantial difference in the running times of the three algorithm was observed. Upon further examination, the pattern system used in Algorithm 2 seems to be the issue. The system was designed to discover more variable expressions without having large multi-word dictionaries, but the frequency with which POS patterns repeat in regular English writing leads to a large number of comparisons and a significant loss in efficiency.

VI. CONCLUSION

The three algorithms have a few drawbacks, notably that n-grams and part-of-speech require dictionaries. The dictionaries need to be created and maintained as new words and colloquialisms come into use online. Also, the dictionaries are restricted to detecting the intent of a single language. Machine learning could be used to improve these dictionaries automatically. Support Vector machines-based algorithm requires preprocessing the data to prepare training and testing datasets. Overall Algorithm 1 that use n-grams was better than the other two algorithms, and over 60% of the results were entirely correct – with another approximate 6% partially correct, showing potential for this method of intent detection. This result confirms the findings of Carlos & Yalamanchi [2] that showed the possibility of intent analysis with an accuracy of above 60%.

VII. FUTURE WORK

Going forward, Algorithms 1 and 3 should be tested on a larger sample of data with reviews and posts regarding multiple industries, and the ratios used to determine the intent type of a sentence or post fine-tuned for more precision. For algorithm 1, automatic update of the dictionary should be investigated. For Algorithm 3, the data structure of the input to the support Vector machines as well as the setting of the support Vector machines' parameters should be investigated. For example we should test the algorithm with using the Gaussian kernel.

ACKNOWLEDGMENT

This research was supported by NSERC through the Engage grant. We thank Kaypok Inc. for supporting this work as an industrial partner. We thank Razieh Niazi from Kaypok Inc. who provided insight and expertise that greatly assisted the research.

REFERENCES

[1] Ardener, E. Social Anthropology and Language. *A. S. A. Monographs, 10*, 1971

[2] Carlos, C. S., & Yalamanchi, M. Intention Analysis for Sales, Marketing and Customer Service. *Proceedings of COLING 2012: Demonstration Paper*, 33-40.

[3] Goldberg, A. B., Fillmore, N., Andrzejewski, D., Xu, Z., Gibson, B., & Zhu, X. May all your wishes come true. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on - NAACL '09. doi:10.3115/1620754.1620793

[4] Ramanand, J., Bhavsar, K., & Pedanekar, N. Wishful thinking: Finding suggestions and 'buy' wishes from product reviews. *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Stroudsburg, PA*.

[5] Wu, X. and He, Z. Identifying wish sentence in product reviews. Journal of Computational Information Systems , 2011, 7(5):1607–1613.