Why don’t you try DiS-BUS? a confirmatory approach for converting short text:

Jongho Im and Taikgun Song

January 27, 2017
Outline

1. Motives
2. Methods
3. Application
4. Discussion
Motives
Background for text analysis

Computational linguistic view (Natural Language Process):
- Processing raw test
- Categorizing and Tagging
- Classification
- Extracting information
- Analyzing sentence structure
- Analyzing the meaning of sentences

Analysis view:
- Topic modeling
- Sentimental analysis
- (topic, word) visualization

Reference
Motives
Motivating example: Restaurant review 1

Table 1: Online restaurant review.

<table>
<thead>
<tr>
<th>Score</th>
<th>Review</th>
</tr>
</thead>
</table>
| 5     | Fast and friendly service.  
        | I had their pork belly eggs benedict and it was amazing!  
        | I’ll be back during my visit to Hawaii for their souffle pancakes. |
| 4     | Not the greatest place in the world  
        | but worth a try if you are in the area.  
        | Guy with the sign up the street gave us a coupon.  
        | It worked out good and we were satisfied. I would go back. |

➤ (Partially) available: user’s nickname or id with some demographic variables, a total number of reviews and helpful reviews.
Motives
Motivating example: Restaurant review 2

- Want to polarize reviews into several sub-dimensions used for restaurant evaluation.
  - Service quality
  - Food quality
  - Environment
  - Price
  - Revisit intention
  - Recommendation
- Survey is a classical tool for conducting this evaluation.
- Can we mimic this work with the online review data?
Motives
Motivating Example: MLS data 1

Table 2: An illustrative example of Ames MLS data. AP: asking price, SP: sold price.

<table>
<thead>
<tr>
<th>AP</th>
<th>SP</th>
<th>Public remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>$100,000</td>
<td>$96,000</td>
<td>All new carpet and paint March 2008. Sellers are Licensed Real Estate Agents in the State of Iowa. Seller providing a 1 year Home Warranty.</td>
</tr>
</tbody>
</table>

Many auxiliary information are available: list date, sold date, address, Br/Ba, square fit, built-in year.
Motives
Motivating Example: MLS data 2

Want to know

1. Do public remarks have additional information explaining asking price?
   - A hedonic model is most popular in analyzing housing price and time on the market (TOM).
   - Incorporate remarks onto to the hedonic model.

2. Features on remarks are related to selling strategy?
   - A Realtor will try to provide information that is good for the house itself.
   - If so, is it related to text position within remarks?
For analysis of the above examples, (may) need to convert the unstructured data to have a structured format.

Table 3: Converted structured data

<table>
<thead>
<tr>
<th>ID</th>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

- Input: Voice record, Image and Text (possibly with emoticon or emoji)
- Output: Structured data presented in Table 3.
Motives
Data converting 2

- Looks similar to survey responses.
  - Survey: Sample (from population) -(Sensor)-Data
  - Online review: User-(Unstructured data-Sensor)-Data

- (intrinsically) unbalanced data.
  - Survey: questions are fixed.
  - Online review: topics are chosen by users.
At the final stage, given $p$-dimensional topics, we have two vectors:

$$t = (t_1, \ldots, t_p),$$

and

$$v = (v_1, \ldots, v_p).$$

- (Topic modeling) $t$ is a vector of topic indicator $t_k$ ($k = 1, \ldots, p$), where $t_{k,i}$ has the value of one if the reviewer $i$ states something about topic $k$ and zero otherwise.

- (Sentimental analysis) $v$ is a vector of values assigned on each topic. If $t_k = 0$, then $v_k$ is not available.
In this presentation, we will

- focus on discussion of topic modelings.
- briefly introduce a latent Dirichlet allocation (LDA) approach popularly used in field.
- introduce an algorithm (DiS-BUS) as the proposed confirmatory topic modeling.
Methods
LDA: definitions

N1 A *word* denoted by $w$ is the basic unit of discrete text data.

N2 A *document* is a sequence of words denoted by $d = (w_1, \ldots, w_m)$, where $w_j$ is the $j$-th word, $j = 1, \ldots, m$.

N3 A *topic* denoted by $t$ is a subject of discussion.

N4 A *corpus* is a collection of $n$ documents denoted by $V = \{d_1, \ldots, d_n\}$. 
Methods
LDA: basic concept

- Proposed by Blei, Ng and Jordan (2003).
- Goal was to automatically discover the topics from a corpus.
- A generative probability model: assume that
  1. Choose $m \sim \text{Poisson}(\xi)$.
  2. Choose $\theta \sim \text{Dirichlet}(\alpha)$.
  3. For each of the $m$ words $w_j$:
     3.1 Choose a topic $t_j \sim \text{Multinomial}(\theta)$.
     3.2 Choose a word $w_j$ from $p(w_j | t_j, \beta)$, a multinomial probability conditioned on the topic $t_j$.
- Represent documents (reviews) as random mixtures over latent topics.

\[
p(w) = \sum_{t} p(w_j | t)p(t),
\]
Methods
LDA : estimation

- **Unigram assumption**
  \[ p(w) = \prod_{j=1}^{m} p(w_j). \]

- **Joint distribution for a fixed document** is
  \[ p(\theta, t, w \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{j=1}^{m} p(t_j \mid \theta)p(w_j \mid t_j, \beta), \]
  where \( p(\theta \mid \alpha) \) is a density function of Dirichlet distribution.

- **Probability of corpus (Independence assumption on documents)**:
  \[ p(V \mid \alpha, \beta) = \prod_{i=1}^{n} \int p(\theta_i \mid \alpha) \prod_{j=1}^{m(d_i)} \sum_{t_{i,j}} p(t_{i,j} \mid \theta_i)p(w_{i,j} \mid t_{i,j}, \beta)d\theta_i, \]
Limitations

1. Topics are identified after parameter estimation.
   - Fix $\rho = 10$ for topic dimensions.
   - Words can be sorted in each dimension.

```r
## top.words:
## 1  "decision" "network" "planning" "learning" "design"
## 2  "learning" "time"   "visual" "networks" "logic"
## 3  "tree"    "networks" "model"   "neural" "search"
## 4  "trees"   "algorithm" "memory" "system"  "learning"
## 5  "classification" "data"  "system" "reinforcement" "systems"
## 1  "learning" "models" "belief" "genetic" "research"
## 2  "search"  "networks" "model"   "search" "reasoning"
## 3  "crossover" "bayesian" "theory" "optimization" "grant"
## 4  "algorithm" "data"  "distribution" "evolutionary" "science"
## 5  "complexity" "hidden" "markov"  "function" "supported"
```

Figure 1. Top selected words for ‘cora’ data in R package \textit{lda}.
Limitations

2. Very difficult to match documents into selected topics.
   - Hard to give some values on each material.

Methods
LDA : extensions

1. Hierarchical Dirichlet Processes
   ▶ Teh et al. (2006)
   ▶ Groups that share the same mixture components.

2. Biterm Topic Model.
   ▶ Yan et al. (2013), Cheng et al. (2014) and Pan et al. (2014).
   ▶ Two words are generated given a topic.

3. CRATS
   ▶ Zhang and Chow (2016)
   ▶ Incorporate latent communities, regions, activities, topics and sentiments from geosocial network data.

However most studies are still basically based on LDA. Same limitations with different degrees.
Methods
Extracting Information

Table 4: Locations data

<table>
<thead>
<tr>
<th>OrgName</th>
<th>Location Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Omnicom</td>
<td>New York</td>
</tr>
<tr>
<td>DDB Needham</td>
<td>New York</td>
</tr>
<tr>
<td>Kaplan Thaler Group</td>
<td>New York</td>
</tr>
<tr>
<td>BBDO South</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Georgia-Pacific</td>
<td>Atlanta</td>
</tr>
</tbody>
</table>

Example presented in Natural Language Toolkit (NLTK) in Python.

- Input: key words or target object, OrgName.
- Output: matched data corresponding to input values, Location Name.

Limitations
- Output quality depends on quality of input information.
- Hard to give scores on the matched contents.
Methods
CTM: Confirmatory topic modeling

Want to

▶ keep meaning structure for both topic modeling and sentimental analysis.
▶ assign topic on each sentence (clause).
▶ make some values (Likert scale) on topic assigned sentences clauses.

Propose a confirmatory topic modeling:

▶ assume a certain relationship between clauses and topics.
▶ Similar to confirmatory factor analysis.
A document is a sequence of words or a set of clauses denoted by $d = (w_1, \ldots, w_m)$ or $d = (c_1, \ldots, c_s)$, where $w_j$ is the $j$-th word, $j = 1, \ldots, m$ and $c_k$ is the $k$-th clause, $k = 1, \ldots, s$.

A clause denoted by $c$ is a sequence of words that includes both a subject and a predicate.

An object word denoted by $o$ is a noun or a noun phrase used for subject, object or complement in a clause.

An evaluation word denoted by $e$ is a set of adjective, adverb or verb that is used to describe or predicate an object word within a clause.
CTM: Assumption 1

A1 Each clause can be approximated a set of bi-terms that consists of an object word (phrase) $o$ and an evaluation word $e$,

$$p(c) = p\{(o_1, e_1), \ldots, (o_l, e_l)\}.$$ 

A2 One clause has only one topic.

Biterms consist of two types: (1) biterms exactly associated with a topic and (2) biterms used as predicative purpose in a clause.

A3 Clauses within a document are conditionally independent given a topic. For a fixed topic $t_k (k = 1, \ldots, p)$,

$$p(c_1, c_2 \mid t_k) = p(c_1 \mid t_k)p(c_2 \mid t_k).$$
Methods
CTM: Assumption 2

1. (A1)
   - $s_1 \sim s_2 \rightarrow c_1 \sim c_2$.
   - $c_1 \sim c_2 \rightarrow s_1 \sim s_2$?
   - Ex) “It was bad food” vs "It was good food".

2. (A2) Examples
   - “Pork was super delicious” $\rightarrow$ (delicious, pork) and (super, delicious).

3. (A3) Representation

![Figure 2. Graph representation for CTM](image)

Figure 2. Graph representation for CTM
Methods
CTM: comparison with LDA

Figure 3. Graph representation for LDA

- LDA: Focus on $p(w \mid t_k)$. All words are associated with all topics.
- CTM: Focus on $p(t_k \mid c)$ and it has binary values 1 or 0.
Methods

CTM: DiS-BUS 1

One way to implement CTM is DiS-BUS:

1. Disintegrating
   ▶ Approximate clauses with a set of bitersms.
   ▶ Require a parsing procedure: Google Syntaxnet, NLTK.

2. Straining
   ▶ First select common expression (CE) based on frequency of selected biterms obtained during the ‘disintegrating’ process.

3. Bagging
   ▶ Construct bag-of-biterms with CE corresponding to topic dimensions.
   ▶ As initial, this can be manually implemented.

4. Up-cycling
   ▶ Find additional biterms which will be inserted into bag-of-biterms among the biterms which were not selected as CE.

5. Scoring
Method

CTM: DiS-BUS 2

Figure 4. Flow chart for DiS-BUS
Methods

CTM: Disintegrating 1

(1) Disintegrating (Parsing)

a. Separate sentences into (meaningful) clauses (Most important but most difficult part)
   ▶ Cleaning process is implemented.
   ▶ Difficulties in abbreviation, ‘comma’, ‘period’, ‘and’ and ‘but’.

b. Apply a parsing/tagging algorithm on each clause.
   ▶ Google syntaxnet (Tensorflow or Python) or NPTK (Python).
   ▶ Part-Of-Speech (POS) are tagged.

Figure 5. An example of Google Syntaxnet
Methods
CTM: Disintegrating 2

(1) Disintegrating (Parsing)

b. Apply a parsing/tagging algorithm on each clause.

"the food was very good"

   good JJ ROOT
   +– food NN nsubj
       |   +– the DT det
       +– was VBD cop
       +– very RB advmod

c. Selected biterms which consist of object word and evaluation word: mainly combination of (NN,JJ) or (NN, VB), (NN, RB) (VB,JJ) (VB,RB).

   (food, good) and (very, good)

   ▶ (food, good) is related to topic dimension.
   ▶ (very, good) is a predicative biterm.
CTM: Straining

(2) Straining: select a set of biterms which are often occurred across all reviews.

<table>
<thead>
<tr>
<th>Word 1</th>
<th>Word 2</th>
<th>Word 1</th>
<th>Word 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>excellent</td>
<td>service</td>
<td>find</td>
<td>hard</td>
</tr>
<tr>
<td>fresh</td>
<td>fish</td>
<td>fresh</td>
<td>ingredients</td>
</tr>
<tr>
<td>friendly</td>
<td>service</td>
<td>friendly</td>
<td>staff</td>
</tr>
<tr>
<td>go</td>
<td>back</td>
<td>go</td>
<td>not</td>
</tr>
<tr>
<td>good</td>
<td>food</td>
<td>good</td>
<td>service</td>
</tr>
<tr>
<td>good</td>
<td>value</td>
<td>great</td>
<td>food</td>
</tr>
<tr>
<td>great</td>
<td>service</td>
<td>great</td>
<td>price</td>
</tr>
<tr>
<td>happy</td>
<td>hour</td>
<td>love</td>
<td>place</td>
</tr>
<tr>
<td>made</td>
<td>home</td>
<td>nice</td>
<td>restaurant</td>
</tr>
<tr>
<td>quick</td>
<td>service</td>
<td>reasonable</td>
<td>price</td>
</tr>
<tr>
<td>recommend</td>
<td>highly</td>
<td>recommend</td>
<td>place</td>
</tr>
<tr>
<td>recommend</td>
<td>restaurant</td>
<td>slow</td>
<td>service</td>
</tr>
</tbody>
</table>

- Degree of common expression is determined by researcher.
- Can be viewed as training sample.
Methods
CTM: Bagging

(3) Bagging

- Assign topics on each biterm with external information.
- Food (F), Service(S), Price (P) Environment (E), Revisit (Rv), and Recommendation (Re).

Table 6: Topic assigned common expression

<table>
<thead>
<tr>
<th>Word 1</th>
<th>Word 2</th>
<th>Topic</th>
<th>Word 1</th>
<th>Word 2</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>excellent</td>
<td>service</td>
<td>S</td>
<td>find</td>
<td>hard</td>
<td>E</td>
</tr>
<tr>
<td>fresh</td>
<td>fish</td>
<td>F</td>
<td>fresh</td>
<td>ingredients</td>
<td>F</td>
</tr>
<tr>
<td>friendly</td>
<td>service</td>
<td>S</td>
<td>friendly</td>
<td>staff</td>
<td>S</td>
</tr>
<tr>
<td>go</td>
<td>back</td>
<td>Rv</td>
<td>go</td>
<td>not</td>
<td>Rv</td>
</tr>
<tr>
<td>good</td>
<td>food</td>
<td>F</td>
<td>good</td>
<td>service</td>
<td>S</td>
</tr>
<tr>
<td>good</td>
<td>value</td>
<td>P</td>
<td>great</td>
<td>food</td>
<td>F</td>
</tr>
<tr>
<td>great</td>
<td>service</td>
<td>S</td>
<td>great</td>
<td>price</td>
<td>P</td>
</tr>
<tr>
<td>happy</td>
<td>hour</td>
<td>P</td>
<td>love</td>
<td>place</td>
<td>E</td>
</tr>
<tr>
<td>made</td>
<td>home</td>
<td>E</td>
<td>nice</td>
<td>restaurant</td>
<td>E</td>
</tr>
<tr>
<td>quick</td>
<td>service</td>
<td>S</td>
<td>reasonable</td>
<td>price</td>
<td>P</td>
</tr>
<tr>
<td>recommend</td>
<td>highly</td>
<td>Re</td>
<td>recommend</td>
<td>place</td>
<td>Re</td>
</tr>
<tr>
<td>recommend</td>
<td>restaurant</td>
<td>Re</td>
<td>slow</td>
<td>service</td>
<td>Service</td>
</tr>
</tbody>
</table>
(4) Up-cycling

- Why up-cycling?
  - Not enough to cover all clauses with CE.
  - Hierarchical structure on words.
    Ex) “great ahi" is essentially same with “great food".
- Can use online dictionary.
  - (JJ, VB, RB) can be compared using a thesaurus.
  - NN can be compared using a classical dictionary.

\[(good, food) \sim (great, ahi)\]

- Can incorporate some machine learning algorithm on clauses level.
- Need to consider a trade-off between the size of upcycling and accuracy.
(5) Scoring

▶ Require another dimension or work.
▶ A naive way can be directly applied into a certain online review.
  ▶ Overall user’s scores are available.
  ▶ Assign positive value on biterms obtained from users, who rated 5 or 4 score for the product or service.
  ▶ Assign negative value on biterms obtained from users, who rated 1 or 2 score for the product or service.
▶ Pang et al. (2002): about 80% accuracy in a sentimental analysis for movie reviews. Use unigram with POS (JJ).
Application

TripAdvisor data 1

- Hawaii restaurants (which have enough reviews.)
- Reviews on the first page are collected using a tailored scraping method.
- About 7,000 reviews.
- Randomly select 200 reviews for illustrative purpose.
Figure 6. A cumulative function of unigrams.

- Words (unigrams) are ordered with frequency.
Table 7: Example of LDA with 6 topic categories.

- Fix $p = 6$ for topic dimensions.
- Top 10 words are selected for each topic dimension.
- Topic specification is difficult. See ‘food’.
<table>
<thead>
<tr>
<th>Food</th>
<th>Service</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>eat</td>
<td>fast</td>
<td>affordable</td>
</tr>
<tr>
<td>enjoyable</td>
<td>also</td>
<td>perfect</td>
</tr>
<tr>
<td>excellent</td>
<td>amazing</td>
<td>charge</td>
</tr>
<tr>
<td>fabulous</td>
<td>attentive</td>
<td>enough</td>
</tr>
<tr>
<td>ahi</td>
<td>awful</td>
<td>fare</td>
</tr>
<tr>
<td>ahi</td>
<td>make</td>
<td>generous</td>
</tr>
<tr>
<td>amazing</td>
<td>nice</td>
<td>money</td>
</tr>
<tr>
<td>apple</td>
<td>not</td>
<td>offer</td>
</tr>
<tr>
<td>ate</td>
<td>order</td>
<td>pay</td>
</tr>
<tr>
<td>strawberries</td>
<td>told</td>
<td>price</td>
</tr>
<tr>
<td></td>
<td></td>
<td>save</td>
</tr>
</tbody>
</table>
Table 9: Bag-of-Biterms (Upcycling result 2)

<table>
<thead>
<tr>
<th>Environment</th>
<th>Revisit</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>clean convention</td>
<td>definitely return</td>
<td>books recommend</td>
</tr>
<tr>
<td>clean convention</td>
<td>return sushi</td>
<td>definitely recommend</td>
</tr>
<tr>
<td>good place</td>
<td>again come</td>
<td>hotel recommend</td>
</tr>
<tr>
<td>convention location</td>
<td>also go</td>
<td>factor recommend</td>
</tr>
<tr>
<td>convention location</td>
<td>go</td>
<td>recommend</td>
</tr>
<tr>
<td>convention location</td>
<td>back come</td>
<td>really recommend</td>
</tr>
<tr>
<td>convention location</td>
<td>really</td>
<td>recommend</td>
</tr>
<tr>
<td>able location</td>
<td>go</td>
<td>recommend</td>
</tr>
<tr>
<td>attractive room</td>
<td>again return</td>
<td>recommend</td>
</tr>
<tr>
<td>beach fabulous</td>
<td>littler return</td>
<td>shrimp</td>
</tr>
<tr>
<td>beautiful lobby</td>
<td>many return</td>
<td>thoroughly</td>
</tr>
<tr>
<td>good</td>
<td>go views</td>
<td>totally</td>
</tr>
<tr>
<td>good</td>
<td></td>
<td>well</td>
</tr>
</tbody>
</table>
1. Im, Song, Lee and Kim. Confirmatory topic modeling for short text.


3. Seo, Im and Mikelbank. A reduced hedonic model incorporating bargaining power.

4. Im, Seo and Cetin. Is house energy efficiency cost passed to renters? Evidences from rental apartments posting in Craigslist.

- New type of data + existing method → new story.
- New type of data ← new (statistical) methodology
Discussion

1. Proposed a confirmatory topic modeling approach which enables to handle short text.
2. An algorithm (DiSBUS) is suggested to achieve CTM.
3. Worked well for TripAdvisor data and also other online review data such as Craigslist postings and Zillow postings.
4. An evaluation method should be designed to check the performance.