Natural Language Processing with E-Commerce Review

HUANG Wenjie wjhuang6-c@my.cityu.edu.hk
Department of Information Systems
City University of Hong Kong
Tat Chee Avenue, Kowloon, Hong Kong SAR

HU Wei weihu24-c@my.cityu.edu.hk
Department of Information Systems
City University of Hong Kong
Tat Chee Avenue, Kowloon, Hong Kong SAR

Abstract

Keywords: Machine Learning, Portfolio, Support Vector Machine, Decision Tree, Random Forest, Boosting

1. Introduction

As one example of a topic model, latent Dirichlet allocation (LDA) is a generative probabilistic model of a corpus. The basic idea of LDA is that each document is represented as random mixtures over latent topics and each topic is characterized by a distribution over words. Since each document is essentially a distribution over topics, rather than a single topic, LDA model is called an admixture mixture or mixed membership model [Erosheva et al. (2004)]. This model has many other applications beyond text analysis, e.g., genetics [Pritchard et al. (2000)], health science [Erosheva et al. (2007)], social network analysis [Airoldi et al. (2008)].

In this project, we will employ an LDA model to analyze customer reviews collected from a shopping website. The dataset in total includes 23,486 records of reviews. For each review, we get information like clothing id, reviewer's age, review title, review text and so on. To conduct text analysis, we first do the pre-processing and vectorization on the text to transform text information into numerical features. The next step is to calculate Term Frequency-Inverse Document Frequency (TF-IDF). Then we train an LDA model and an LSA (Latent Semantic Analysis) model to extract latent topics, calculate text correlation and output most relevant results. Finally, we do cross-validation of the topic model. The following sections present the detailed process of our project.

2. Experiment Design

In this experiment, after preprocessed the data, we calculate the TF-IDF for each review and take use of LSA and LDA to analyze the topics of the reviews. And we use cross-validation to select the optimal number of topics for the topic models. Then we predict the sentiment (Recommended vs Not Recommended) of reviews using the bag-of-topic features (Unigram, Bigram and Trigram) using Logistic Regression.

2.1 Dataset

Women's Clothing E-Commerce dataset revolving around the reviews written by customers. This dataset includes 23,486 rows and 10 feature variables. Each row corresponds to a customer review, and includes the variables:

- Clothing ID: Integer Categorical variable that refers to the specific piece being reviewed.
- Age: Positive Integer variable of the reviewers age.
- Title: String variable for the title of the review.
- Review Text: String variable for the review body.
- Rating: Positive Ordinal Integer variable for the product score granted by the customer from 1 Worst, to 5 Best.
- Recommended IND: Binary variable stating where the customer recommends the product where 1 is recommended, 0 is not recommended.
- Positive Feedback Count: Positive Integer documenting the number of other customers who found this review positive.
- Division Name: Categorical name of the product high level division.
- Department Name: Categorical name of the product department name.
- Class Name: Categorical name of the product class name.

2.2 Preprocessing

For textual data, we have to transform the unstructured texts into some numerical data which can be feed to machine learning algorithms. And the preprocessing step is to make the text cleaner and easier to be processed in the following steps. We assume the smallest unit of information in the texts is word in this experiment rather than characters. And the texts can be represented as word sequences. We follow the following steps to preprocess the texts:

- Text tokenization: Text can be regarded as list of word in English scenario and we can split the sentences into words by whitespace. Then we convert all the words into lower case;
- Removing stopwords: Stop words are those words that do not contribute to the deeper meaning of the phrase. We can remove the stopwords in our experiment;
- Removing English punctuations: The punctuation like commas and quotes can also be removed.
- Text stemming: Stemming is the process of reducing the inflectional forms or derivationally related forms of a word to a common base form. We take use of Porter stemmer here for the text stemming;
- Removing low frequency words: We remove all the low frequency words such as the words that occur only once.

Table 1: TF-IDF weighted feature vector of the sample review

Word	Occurence	TF-IDF
absolut	1	0.3737423347857016
comfort	1	0.2244287454373877
sexi	1	0.5253917570308515
silki	1	0.5852463877033008
wonder	1	0.4374912258944229

After the preprocessing, we get the vector of textual features (single word, unigram). And we can also take use of ngrams to get the vector of bigram and trigram textual features.

2.3 TF-IDF and Topic Model

We can reweight the vector of textual features using Term Frequency-Inverse Document Frequency (TF-IDF). TF means term-frequency, which is the number of times a term occurs in a given document. And TF-IDF means term-frequency times inverse document-frequency: $tf - idf(t, d) = tf(t, d) \times idf(t)$. Here, $idf(t) = log\frac{1+n_d}{1+df(d,t)} + 1$, where n_d is the total number of document, $df(d, t)$ is the number of documents containing term t. We regard each review as a document. Then the corpus of documents can thus be represented by a matrix with one row per document and one column per token (e.g. word) occurring in the corpus.

Then we take use of Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA) for the topic analysis. LDA assumes documents are related to a set of topics and these topics relate to a set of words. It is a matrix decomposition technique on the document-term matrix. LSA learns latent topics by performing a matrix decomposition (SVD) on the term-document matrix.

2.4 Classification and Sentiment Analysis

We regard the Recommend IND (Recommended/Not Recommended) as the binary sentiment score (0 and 1). And taking use of bag-of-topic features to predict the sentiment of each review, which is and traditional classification problem in machine learning. We use Logistic regression classifier here.

3. Result

In this section, we are going to discuss the results of our topic model and sentiment prediction.

3.1 Preprocessing

The sample review before process is **"Absolutely wonderful - silky and sexy and comfortable"**. And after preprocessing, the vector of textual features is [**'absolut', 'wonder', 'silki', 'sexi', 'comfort'**]. And after reweighting the vector using TF-IDF, we get the result as shown in Table 1. We can see that **"silki"** and **"sexi"** are with highest TF-IDF weight.

3.2 LSA

We use gensim.models.LsiModel to build the LSA model with 100 topics. In order to demonstrate the topics of the LSA model, we map query word **"Good dress"**, an intuitively positive review, to the 100 dimensional topic space with LSI model. The result is shown in Table 2.

Table 2: 100 dimensional topic space with LSI model of the query "Good dress"

Num	Coordinate	Num	Coordinate	Num	Coordinate	Num	Coordinate	Num	Coordinate
0	0.26	20	-0.006	40	-0.179	60	-0.181	80	-0.012
1	-0.006	21	-0.114	41	-0.017	61	-0.024	81	-0.169
2	0.713	22	0.123	42	-0.133	62	0.117	82	-0.138
3	-0.225	23	-0.27	43	-0.295	63	0.089	83	-0.149
4	0.128	24	-0.084	44	0.022	64	0.154	84	-0.042
5	-0.056	25	-0.003	45	-0.04	65	-0.073	85	0
6	-0.01	26	0.026	46	-0.022	66	0.06	86	-0.013
7	-0.192	27	0.035	47	0.072	67	-0.095	87	0.032
8	-0.126	28	0.148	48	-0.092	68	0.013	88	0.088
9	-0.065	29	0.132	49	0.028	69	0.07	89	-0.074
10	0.129	30	-0.017	50	-0.043	70	-0.112	90	-0.091
11	0.184	31	0.002	51	0.05	71	-0.004	91	0
12	-0.004	32	0.106	52	0.025	72	0.022	92	-0.028
13	-0.031	33	0.049	53	-0.068	73	0.069	93	0.177
14	-0.08	34	-0.051	54	-0.011	74	0.173	94	0.003
15	-0.03	35	0.25	55	-0.133	75	-0.066	95	0.081
16	-0.127	36	0.096	56	-0.148	76	-0.109	96	-0.1
17	-0.139	37	0.019	57	0.046	77	-0.15	97	-0.079
18	0.025	38	-0.007	58	-0.105	78	-0.034	98	0.045
19	0.094	39	0.114	59	0.09	79	-0.037	99	-0.006

Then we can calculate the cosine similarity/correlation degree between documents and query word and the sorted results of top 10 correlated reviews are shown in Table 3.

Table 3: Top 10 documents that is most correlated to the query "Good dress"

No.	No. of Doc	Similarity	Review
1	8111	0.7012306	This dress is adorable. dress it up or dress it down
2	4102	0.69777906	This is awesome multi-season dress.
3	14947	0.69364554	I love the swing and the pretty color of the dress. it's fun to dress up or dress down. the fabric is a little thin so you will need good undergarments if you have lumps and bumps.
4	12908	0.69123614	Horrible fit. i do not understand why they but a aline dress with a skin non aline camisole under the dress.

5	6295	0.6826043	This dress is a good casual summer dress. the material is thin and feels nice on. the dress was very wrinkled when it arrived but that came out very easily and the fabric doesn't really wrinkle easily after initial wash and steam. the color is very vibrant and the fit is loose. however
6	6970	0.6715683	Good quality; casual feel good dress. it can be worn as dress
7	4014	0.6701734	I love this dress . \r\nperfect fit and very good quality.
8	8523	0.66510737	As the previous reviewer mentioned
9	10193	0.6491452	I fist saw this dress in the window in alexandria
10	12380	0.6408222	Really cute stress! i love the pattern and feel that this dress you can easily dress up or dress down. the neckline is the main reason why i bought the dress. it is a dress that hits above my knees so i like how modest it is. the slightly open back is also a nice touch. it's a dress that can easily be dressed up or dressed down. perfect spring dress!

3.3 LDA

We use `gensim.models.LdaModel` to build the LDA model with 100 topics. We use Perplexity and Coherence Score to evaluate this model. The perplexity is **-16.616** and the coherence score is **0.371**. In order to demonstrate the topic model, we use `pyLDAvis` package for interactive topic model visualization and get the *html* file `lda_ntopic=100.html` as shown in Figure 1. The left side display the intertopic distance map and the right side display the top-30 most relevant terms for the topic.

3.4 Sentiment Prediction

In the Figure 3 you can find the histogram of sentiment of each review. 1 represents positive sentiment (Recommended) and 0 represents negative sentiment (Not Recommended). We can see that most of the reviews are positive.

We use `ELI5` to get the ordered list of relevant textual features to positive sentiment (Recommended) as shown in *html* file `sentiment_topfeature.html` and `eli5_predict.html` (see Figure 2). The green colour represent the term relevant to positive sentiment and the red color represents the term relevant to negative sentiment.

We use cross-validation to select the topic number and with minimum cross-validation error, we get the best number of topics is 90. The cross-validation error plot is shown in Figure 4. The details can be found in Table 7 in Appendix.

3.5 Bigram and Trigram

We use ngrams model to get the bigram and trigram corpus and rerun the cross-validation and sentiment prediction using logistic regression. First of all, we get the table of occurrence of grams as shown in Table 4.

Then we can get the plots of cross-validation error of bigram and trigram corpus as shown in Figure 5-6. The details can be found in Table 7 in Appendix.

Figure 1: Screenshot of pyLDAvis of generated LDA model

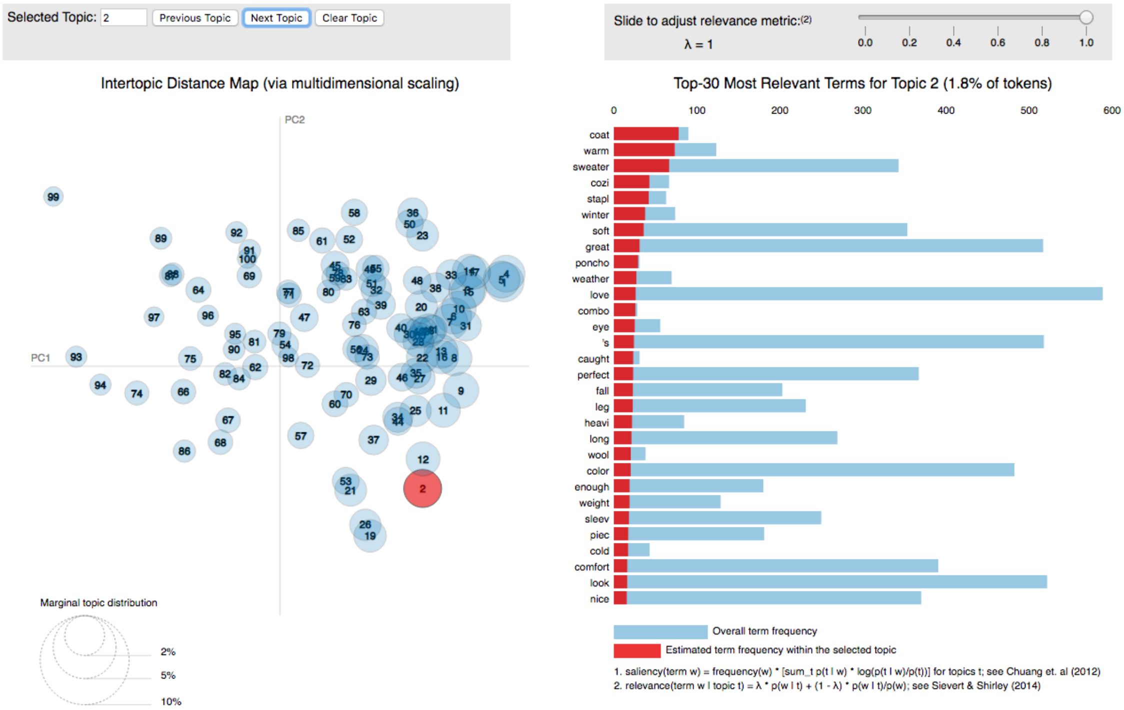
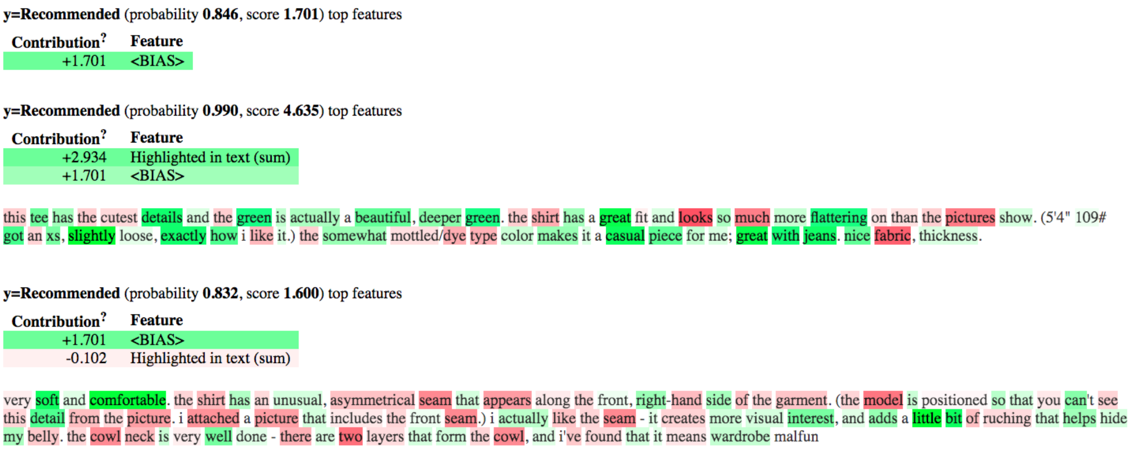


Figure 2: Screenshot of Eli5 of prediction of sentiment



y=Recommended (probability **0.846**, score **1.701**) top features

Contribution?	Feature
+1.701	<BIAS>

y=Recommended (probability **0.990**, score **4.635**) top features

Contribution?	Feature
+2.934	Highlighted in text (sum)
+1.701	<BIAS>

this tee has the cutest details and the green is actually a beautiful, deeper green. the shirt has a great fit and looks so much more flattering on than the pictures show. (5'4" 109# got an xs, slightly loose, exactly how i like it.) the somewhat mottled/dye type color makes it a casual piece for me; great with jeans. nice fabric, thickness.

y=Recommended (probability **0.832**, score **1.600**) top features

Contribution?	Feature
+1.701	<BIAS>
-0.102	Highlighted in text (sum)

very soft and comfortable. the shirt has an unusual, asymmetrical seam that appears along the front, right-hand side of the garment. (the model is positioned so that you can't see this detail from the picture. i attached a picture that includes the front seam.) i actually like the seam - it creates more visual interest, and adds a little bit of ruching that helps hide my belly. the cowl neck is very well done - there are two layers that form the cowl, and i've found that it means wardrobe malfun

3.6 Conclusion

We find that the best model for sentiment prediction is logistic regression using bag-of-topic features generated by LDA model with **number of topics of 90** using **unigram corpus** with cross-validation error of 17.66%.

Acknowledgments

Figure 3: Histogram of review sentiment

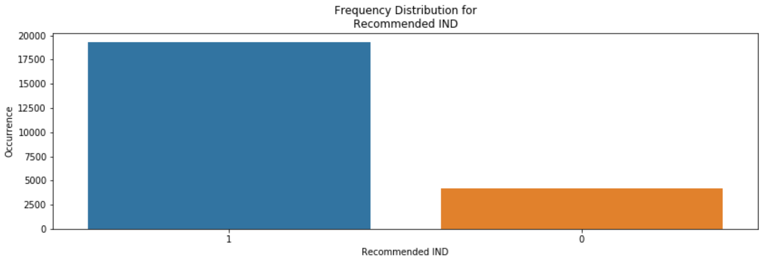
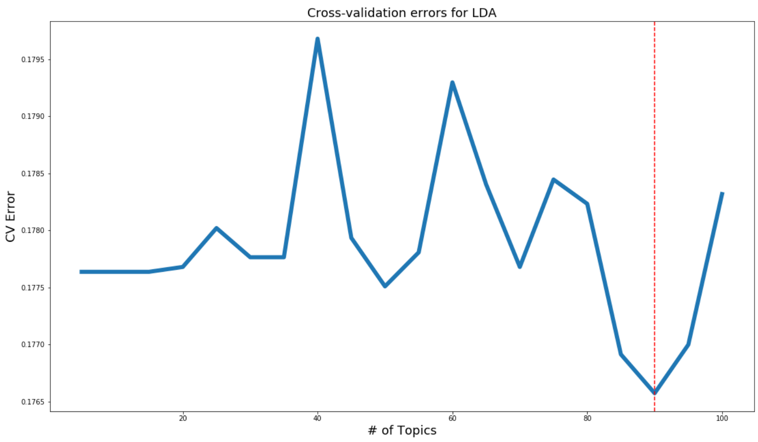


Figure 4: Cross-validation error plot of unigram LDA model

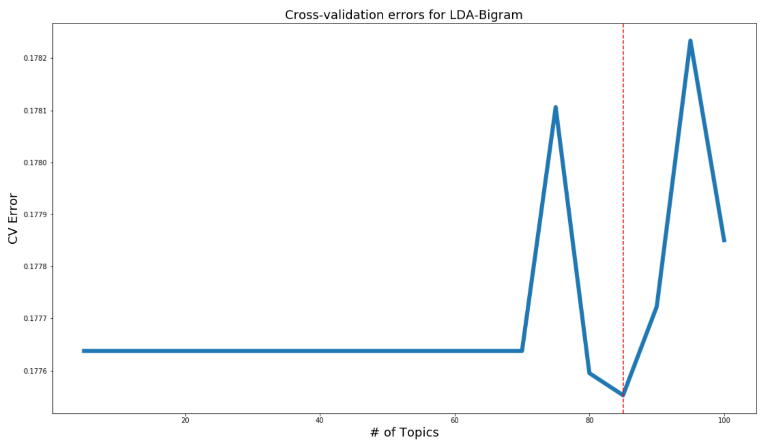


We would like to acknowledge support for Ms. LUO Linbin and Dr. WU Jing.

Table 4: Grams table of review corpus

	1-Gram	Occurrence	2-Gram	Occurrence	3-Gram	Occurrence
0	look	2412	look like	479	realli want love	72
1	dress	2132	go back	273	want love dress	67
2	like	1958	want love	254	realli want like	42
3	fit	1715	realli want	162	fit true size	40
4	top	1713	love dress	136	made look like	30
5	size	1594	made look	136	want love top	29
6	love	1474	5 4	113	make look like	28
7	would	1348	make look	111	look like matern	28
8	fabric	1267	felt like	109	sadli go back	28
9	color	1061	usual wear	104	look like wear	27
10	back	1039	true size	104	order usual size	26
11	wear	1026	run small	99	way much fabric	25
12	order	997	love color	97	like matern top	24
13	small	914	fit well	97	usual wear size	24
14	return	901	feel like	96	one go back	21
15	5	871	size small	95	look noth like	20
16	realli	869	look great	91	dress look like	19
17	tri	805	much fabric	90	first time wore	18
18	materi	750	5 5	88	would look good	17
19	shirt	732	look good	87	go back love	17

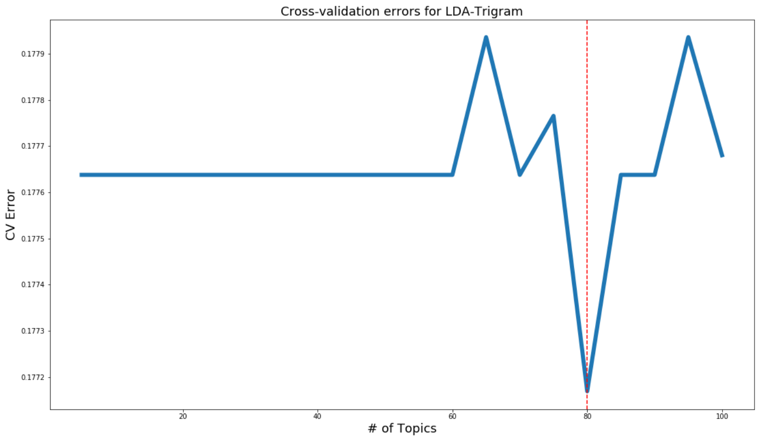
Figure 5: Cross-validation error plot of bigram LDA model



References

Edoardo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9(Sep):1981–2014, 2008.

Figure 6: Cross-validation error plot of trigram LDA model



Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101(suppl 1): 5220–5227, 2004.

Elena A Erosheva, Stephen E Fienberg, and Cyrille Joutard. Describing disability through individual-level mixture models for multivariate binary data. *The annals of applied statistics*, 1(2):346, 2007.

Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.

Appendix A.

- lda_ntopic=100.html
- eli5_topfeature.html
- eli5_predict.html

Table 5: Top 10 Topics generated by LSI model

No.	Topic
0	0.196*”dress” + 0.157*”top” + 0.156*”size” + 0.146*”love” + 0.142*”fit” + 0.136*”’s” + 0.135*”great” + 0.135*”look” + 0.126*”wear” + 0.126*”n’t”
1	0.306*”great” + 0.255*”jean” + 0.228*”comfort” + -0.222*”small” + -0.176*”size” + 0.159*”soft” + -0.154*”order” + -0.148*”larg” + 0.144*”love” + 0.131*”sweater”
2	0.743*”dress” + -0.316*”shirt” + -0.237*”top” + -0.142*”sweater” + -0.115*”cute” + 0.097*”perfect” + 0.097*”beauti” + -0.095*”jean” + 0.081*”flatter” + -0.073*”sleev”
3	0.261*”size” + 0.239*”small” + 0.217*”jean” + -0.207*”shirt” + 0.204*”pant” + 0.183*”run” + 0.167*”medium” + -0.163*”dress” + 0.141*”order” + 0.136*”usual”
4	0.407*”shirt” + -0.266*”sweater” + 0.211*”run” + 0.204*”top” + 0.191*”larg” + 0.179*”dress” + 0.163*”small” + -0.156*”skirt” + -0.155*”jean” + -0.155*”pant”
5	0.676*”sweater” + -0.255*”shirt” + -0.183*”pant” + -0.182*”jean” + -0.141*”waist” + -0.139*”skirt” + 0.126*”beauti” + 0.117*”medium” + 0.116*”warm” + -0.115*”short”
6	0.510*”shirt” + -0.424*”top” + -0.158*”great” + -0.146*”nice” + 0.139*”compliment” + 0.135*”tri” + 0.130*”store” + -0.122*”fabric” + 0.121*”petit” + -0.121*”beauti”
7	0.396*”top” + -0.336*”shirt” + -0.236*”sweater” + 0.226*”compliment” + -0.167*”dress” + 0.149*”mani” + -0.139*”great” + -0.135*”soft” + -0.126*”run” + 0.125*”receiv”
8	0.261*”perfect” + -0.253*”cute” + -0.238*”pant” + -0.205*”run” + 0.200*”length” + 0.196*”top” + 0.180*”petit” + -0.168*”super” + 0.151*””” + 0.146*”xs”
9	0.430*”skirt” + -0.354*”cute” + -0.257*”super” + 0.227*”shirt” + 0.221*”beauti” + -0.210*”top” + 0.191*”size” + 0.169*”color” + 0.157*”true” + 0.150*”qualiti”
10	-0.648*”skirt” + -0.248*”compliment” + 0.224*”pant” + -0.160*”’s” + 0.151*”color” + -0.148*”mani” + -0.128*”cute” + -0.118*”waist” + -0.109*”wore” + -0.105*”receiv”

Table 6: Top 10 Topics generated by LDA model

No.	Topic

0	0.068*”purpl” + 0.054*”everyon” + 0.051*”5’8” + 0.048*”34d” + 0.046*”wish” + 0.044*”requir” + 0.039*”fenc” + 0.036*”difficult” + 0.032*”came” + 0.032*”guess”
1	0.036*”hourglass” + 0.024*”choic” + 0.020*”ship” + 0.018*”figur” + 0.016*”funki” + 0.016*”10.” + 0.014*”vivid” + 0.014*”discov” + 0.013*”8” + 0.011*”moment”
2	0.031*”eye” + 0.027*”gone” + 0.026*”caught” + 0.024*”dot” + 0.022*”suggest” + 0.015*”post” + 0.014*”anyth” + 0.014*”babi” + 0.013*”hair” + 0.012*”store”
3	0.056*”transit” + 0.051*”next” + 0.045*”winter” + 0.041*”season” + 0.037*”live” + 0.036*”mute” + 0.034*”warm” + 0.033*”velvet” + 0.028*”unlin” + 0.027*”fall”
4	0.061*”roll” + 0.049*”remind” + 0.042*”trouser” + 0.039*”sturdi” + 0.038*”featur” + 0.038*”3/4” + 0.032*”plenti” + 0.019*”hippi” + 0.018*”6-8” + 0.018*”sleev”
5	0.036*”scratchi” + 0.034*”event” + 0.033*”keeper” + 0.031*”option” + 0.029*”versatil” + 0.028*”shop” + 0.026*”us” + 0.021*”past” + 0.021*”luck” + 0.019*”weigh”
6	0.025*”romper” + 0.019*”5’9” + 0.017*””” + 0.016*”length” + 0.016*”longer” + 0.015*”knee” + 0.015*”120” + 0.014*”hit” + 0.013*”’m” + 0.013*”eleg”
7	0.045*”pleas” + 0.036*”mind” + 0.033*”uncomfort” + 0.032*”lower” + 0.024*”garment” + 0.023*”flair” + 0.021*”jersey” + 0.020*”fuller” + 0.015*”debat” + 0.013*”fair”
8	0.030*”upper” + 0.020*”packag” + 0.014*”pleat” + 0.013*”challeng” + 0.013*”12.” + 0.013*”sz” + 0.011*”arm” + 0.011*”dryer” + 0.011*”soon” + 0.011*”fleec”
9	0.046*”mine” + 0.033*”36c” + 0.027*”crazi” + 0.026*”well-mad” + 0.025*”layer” + 0.024*”alon” + 0.024*”pink” + 0.019*”sold” + 0.018*”floor” + 0.012*”matronli”
10	0.035*”0” + 0.029*”textur” + 0.029*”zip” + 0.023*”rib” + 0.020*”zipper” + 0.017*”stop” + 0.017*”m/l” + 0.016*”stock” + 0.016*”size” + 0.015*”gather”

Table 7: Cross-validation errors using unigram corpus

Num of Topics	CV Error - Unigram	CV Error - Bigram	CV Error - Trigram
5	0.1776	0.1776	0.1776
10	0.1776	0.1776	0.1776
15	0.1776	0.1776	0.1776
20	0.1777	0.1776	0.1776
25	0.178	0.1776	0.1776
30	0.1778	0.1776	0.1776
35	0.1778	0.1776	0.1776
40	0.1797	0.1776	0.1776

45	0.1779	0.1776	0.1776
50	0.1775	0.1776	0.1776
55	0.1778	0.1776	0.1776
60	0.1793	0.1776	0.1776
65	0.1784	0.1776	0.1779
70	0.1777	0.1776	0.1776
75	0.1784	0.1781	0.1778
80	0.1782	0.1776	**0.1772**
85	0.1769	**0.1776**	0.1776
90	**0.1766**	0.1777	0.1776
95	0.177	0.1782	0.1779
100	0.1783	0.1779	0.1777