Empirical comparison of LDA with MG-LDA

Nina Dongen, Jörg Sander and Bram Rodenburg

Abstract

In this paper we replicated the Latent Dirichlet Allocation (LDA) model introduced by Blei *et al.* [1] and additionally replicated an extension of the by now standard LDA model: the Multi-Grain Topic Model as proposed by Titov and McDonald [5]. The standard LDA model is used to allocate topics to documents unsupervised, the downside is that this topic allocation can be too coarse for certain aims, such as the extraction of ratable aspects from a dataset. Our goal was to show that the Multi-Grain LDA (MG-LDA) provides an unsupervised solution to refine topic allocation, by the addition of a second layer of subtopics. Moreover, we investigated the performance of both models on different datasets.

1 Introduction

The world we live in becomes more and more a digital world. Most of us cannot imagine life without internet. The World Wide Web expands every second; it is estimated that the Indexed Web contains at least 4.73 billion pages.[1] The vast majority of this ever growing content is noisy. Therefore, tools to extract relevant information unsupervised from this huge amount of available data is of importance. One area of interest is the area of e-commerce. A growing amount of products is offered and bought online and with that we got accustomed to search and share online information about these products. Online sellers try to collect customer reviews to inform new buyers; while customers are more willing to provide product reviews. The downside of this success is that it can become difficult, even impossible, to read all the reviews of a certain product because there are too many. Therefore, unsupervised mining tools able to interpret customer reviews are of great value.

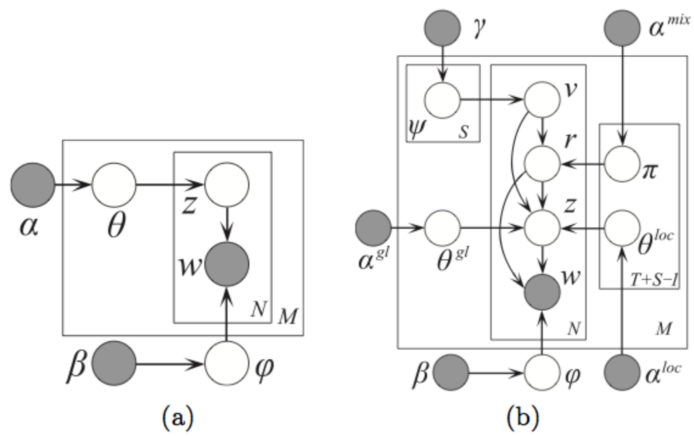
In this paper we will compare two models used to interpret customer reviews. We will start with a replication of a standard Latent Dirichlet Allocation (LDA) model as introduced by Blei *et al.* [1]. This is a model often used for unsupervised topic allocation. Second, we will replicate the Multi-Grain Topic Model (MG-LDA) as proposed by Titov and McDonald [5]. The goal of this model envisioned by Titov and McDonald, is to provide the means for sentiment retrieval. For example, to extract the sentiment about a certain object, such as a specific camera. This Multi-Grain Topic Model is an extension of a standard LDA model.

Our goal is to test both the LDA and the MG-LDA models on three different datasets *software*, *sports* & *outdoors* and *magazines* and study the performance on topic allocation and its usefulness for rating. We expect MG-LDA algorithm to perform better than the standard LDA. Also we expect to get better results from the datasets *software* and *sports* & *outdoors* compared to *magazines* due to complexity of their content.

In the next section we discuss our used methods in three parts. First, we will provide an explanation of the standard LDA model; additionally, we will discuss its limitations and why it may be fruitful to extend it. Second, we explain the workings of the MG-LDA and its possible benefits. Third, we will give a description of Gibbs sampling. In the following section 3, our experimentation and results are displayed. It will explain the used data and the implementation of both LDA and

[1] On Friday the 13^{th} of November 2015; source at www.worldwidewebsize.com

Figure 1: Graphical models of LDA (a) and MG-LDA (b)



MG-LDA models, followed by our results. In section 4 the results will be analyzed. Finally, we end the paper with a short conclusion.

2 Method

2.1 LDA

The LDA model, initially introduced by Blei *et al.* [1], is a generative probabilistic model that can be used for collections of grouped discrete data. In this paper we assume the LDA model to be applied to a corpus consisting of a collection of documents, with words being the basic elements. Generation of a document collection D under the LDA method is as follows:

- for $z = 1...Z$:
  - choose a distribution of words $\varphi_z \sim Dir(\beta)$
- for each word i in document $d \in D$:
  - choose a distribution of topics $\theta_d \sim Dir(\alpha)$
    - choose topic $z_{d,i} \sim \theta_d$
    - choose word $w_{d,i} \sim \varphi_{z_{d,i}}$

Where Z is the number of latent topics; φ_z is the distribution of words in latent topics z; θ_d is the distribution of topics in documents d; $z_{d,i}$ is the topic index for word i in document d; $w_{d,i}$ is the index for word i in document d; α and β are both hyperparameters for the Dirichlet distributions from which the discrete distributions are drawn. We can see a representation of this model in Figure 1(a).[2]

The LDA method represents its documents as a bag-of-words. As a result, co-occurrences of words can only be taken into account at document level; therefore, the distribution of topics for each document is fixed. This means for example that when processing specific product data about tablets, terms are classified into product instances, say the global topics *Ipad Air* and *Asus ZenPad*. But, if your goal is product rating, then calculating the rating per product is not very informative. If we want to be more informative and summarize how these products are rated by users in customer reviews, we need a local, finer grained layer of topics. Say for example local topics as *quality of sound*,

[2]This model was provided by Titov and McDonald in *Modeling online reviews with Multi-Grain Topic Models*, WWW 2008 Beijing, 2008.

user friendliness and *design*, for both the *Ipad Air* and the *Asus ZenPad* to provide comparable information about these specific product features. Extracting this extra layer of topics enables us to compare local subclasses of topics and provides a foundation much richer in information for rating.

The realization that one needs local topics next to global topics for more informative rating was the inspiration for Titov and McDonald [5] to extend the standard LDA model; their approach will be explained in the following section.

2.2 MG-LDA

In contrast to the LDA model, which models just one type of topics, the MG-LDA model produces two different types of topics, namely *global topics* and *local topics*. Similar to LDA, in MG-LDA each document has exactly one global topic distribution. Every word in the document is sampled from the mixture of global topics. However more intuitively, the MG-LDA model allows the distribution of local topics to change over the document. This means that different local topic distributions may be assigned to one single document. The generative model assumes that every word in a document can be sampled from a global or local topic distribution that are specific for the local context of the word (i.e. the segment of the document surrounding that word). In the MG-LDA model this local context is introduced by representing a document as a set of *sliding windows* in contrast to a *bag of words*. Each window covers T adjacent sentences. [3]

Titov and McDonald [5] used this two-level construction with the idea that local topics capture ratable aspects, while global topics capture the properties of the reviewed items. As example they give the following segment from a review about a hotel in London: *"... public transport in London is straightforward, the tube station is about an 8 minute walk ...or you can get a bus for £1.50"*. The excerpt may be seen as a mixture of the local topic *location*; the ratable aspect specific for the local context, based on the words "transport", "walk" and "bus". And of the global topic *London* shared by the entire review, based on the words "London", "tube" and "£".

In the MG-LDA model each sliding window v in document d is tied to a corresponding distribution over local topics denoted as $\theta^{loc}_{d,v}$. In addition each window is associated with a distribution $\pi_{d,v}$ defining the preference for either local or global topics. When a word w is sampled all (overlapping) windows covering the sentence s in which the word occurs are taken into account. The windows itself are chosen based on the categorical distribution ψ_s. Allowing the sliding windows to overlap gives the opportunity to cover a larger domain of co-occurrence (i.e. larger than the occurrence within a sentence) and with that the needed foundation to allocate local topics. The symmetrical Dirichlet prior $Dir(\gamma)$ for the distribution of ψ_s is chosen to control the smoothness of topic transitions.

In a model with K^{gl} global topics and K^{loc} local topics, one first has to decide the K^{gl} word distributions for the global topics φ^{gl}_z from a Dirichlet prior $Dir(\beta^{gl})$ and similarly, the K^{loc} word distributions for the local topics $\varphi^{loc}_{z'}$ from a Dirichlet prior $Dir(\beta^{loc})$.Then, generation of a document collection D under the MG-LDA method is as follows:

- for each document $d \in D$:
  - choose a distribution of global topics $\theta^{gl}_d \sim Dir(\alpha^{gl})$
- for each sentence s in document d:
  - choose a distribution of topics $\psi_{d,s}(v) \sim Dir(\gamma)$
- for each sliding window v of document d:
  - choose a distribution of local topics $\theta^{loc}_{d,v} \sim Dir(\alpha^{loc})$
  - choose distribution of topic preferences $\pi_{d,v} \sim Beta(\alpha^{mix})$
- for each word i in sentence s of document d:
  - choose window $v_{d,i} \sim \psi_{d,s}$
  - choose preference $r_{d,i} \sim \pi_{d,v_{d,i}}$

[3] In the original model of Titov and McDonald a window covered three adjacent sentences.

- if $r_{d,i} = gl$ choose global topic $z_{d,i} \sim \theta_d^{gl}$
- if $r_{d,i} = loc$ choose local topic $z_{d,i} \sim \theta_{d,v_{d,i}}^{loc}$
- choose word $w_{d,i}$ from the word distribution $\varphi_{z_{d,i}}^{r_{d,i}}$

Where $Beta(\alpha^{mix})$ is a prior non-symmetrical distribution to choose between global and local topics. Its distribution is non-symmetrical to allow the regulation of preference for either global or local topics. A representation of this model is shown in Figure 1(b).

2.3 Gibbs sampling

Collapsed Gibbs sampling was used as an inference method in all experiments. The implementation of the LDA model uses the method described in [3]. The MG-LDA implementation uses a slightly modified version of the algorithm which is described in [5]. There have been three approaches to inference with LDA: *Expectation Maximization* (EM) with variational inference [1], EM with expectation propagation [4], and Gibbs sampling [3]. The performance of Gibbs sampling is comparable to the other two methods and is known to be less prone to local minima.

Gibbs sampling belongs to the class of samplings methods known as Markov Chain Monte Carlo (MCMC) and provides a way to sample from a joint distribution in the case when only conditional distributions of each unobserved variable can be computed. In the case of LDA this means that we are trying to sample from the posterior distribution $P(\mathbf{z}|\mathbf{w})$ where $\mathbf{z}$ denotes a vector of assignments of words to an unobserved topic and $\mathbf{w}$ denotes a vector of the individual words in the corpus. Whereas in the MG-LDA model one tries to sample from the joint conditional distribution $P(\mathbf{v}, \mathbf{r}, \mathbf{z}|\mathbf{w})$ where $\mathbf{v}$ denotes the unobserved assignment of each word to one of the sliding windows and $\mathbf{r}$ denotes the assignment of a word to a global or local *context* (hereafter referred to as global/local context).

In order to sample from $P(\mathbf{z}|\mathbf{w})$ respectively $P(\mathbf{v}, \mathbf{r}, \mathbf{z}|\mathbf{w})$ using Gibbs sampling method, the full conditional posterior distribution $P(z_{d,i}|\mathbf{z}^{-d,i}, \mathbf{w})$ resp. $P(v_{d,i}, r_{d,i}, z_{d,i}|\mathbf{v}^{-d,i}, \mathbf{r}^{-d,i}, \mathbf{z}^{-d,i}, \mathbf{w})$ is needed where d, i denotes the i-th word in document d. $\mathbf{z}^{-d,i}$ denotes the topic assignments for all words except for the word in document d at position i. The full derivation of the collapsed Gibbs sampler for LDA and MG-LDA is described in [3] resp. [5] and will not be replicated in this paper.

3 Experiments and results

3.1 Collecting the data

We used the original unprocessed datasets *software*, *sports* $\&$ *outdoors* and *magazines* from the large and freely available Multi-Domain Sentiment Dataset (version 2.0).[4] The whole set was originally collected by Blitzer *et al.* in 2007 [2]. The Multi-Domain Sentiment Dataset consists of product reviews taken from Amazon.com and covers many types of products.

We chose the sets *software* and *sports* $\&$ *outdoors* because these are very similar to the datasets used by Titov and McDonald [5].[5] The reviews in these sets are very suitable to topic allocation, because on the whole these reviews are a direct response to the relevant products. Take for example the sentences *"The program works fine on Windows XP and is easy to use"* or *"If you can find something cheaper buy it, this is not worth 80 bucks"* from *software* and *sports* $\&$ *outdoors* respectively. In contrast, we expect *magazines* to be less suitable for unsupervised topic allocation, because the data is more noisy. It may be said that magazines are more complicated products: many influences are in play, such as style (of writing), author, advertisement, photography, informativeness, political view, and so forth; and all these different influences may change per issue. A reader may comment on a certain personage occurring in a certain article that was inserted in a certain issue. For example the excerpt *"Their article on Gavrilo Princip, the assassin of Archduke Francis Ferdinand, shook me to the core"*.

[4]https://www.cs.jhu.edu/ mdredze/datasets/sentiment/

[5]Titov and McDonald originally used datasets with reviews about *MP3 players* and *hotels and restaurants*

For all the mentioned datasets punctuations were removed; also all stopwords were expelled with the help of the English stopword corpus made available by NLTK. Words shorter than two characters were also discarded. Moreover, all datasets were sentence split and words were tokenized. The properties of the datasets are shown in Table 1. We found that using the NLTK Porter stemmer did not result in more usable topics and therefore we decided not to use stemming.

Table 1: Properties of the used datasets

Domain	Reviews	Sentences	Words	Mean words per review
Software	999	8791	66597	66.7
Sports & outdoors	697	4296	30696	44
Magazines	1546	10192	85465	55.3

3.2 Implementation of models

Both models were implemented by means of *Python 2.7*. The programs are available on github. [6] The pseudocode of the models is outlined in algorithm 1 resp. algorithm 2 on the next page. As mentioned above collapsed Gibbs sampling was used for both models and a chain was run for 500 iterations to produce a sample for each of the experiments. Unfortunately no profound tuning of the parameters of the prior distributions could be performed due to time limitations and performance issues. [7] For the LDA model the hyperparameters α, β were set to 0.1 and 0.005. The MG-LDA model used a value of 0.01 for all parameters of the prior distributions.

Pseudocode **LDA**
Input: List of documents & words per document, number of topics, number of Gibbs iterations
Output: word distributions over topics, topic distributions over documents

```
/* Initialize assignment of words in documents to topics        */
InitializeCounters()

for each document d do
    for each word w_d in document d do
        w_d.topic ← randomly selected topic
        UpdateCounters(w_d, w_d.topic)

/* Gibbs sampling iterations                                    */
while iterations ≤ Number of Gibbs iterations do
    for each document d do
        for each word w_d in document d do
            DecreaseCounters(w_d, w_d.topic)
            new_topic ← Sample from P(topic|words, α, β)
            IncreaseCounters(w_d, new_topic)
    iterations++
```

Algorithm 1: LDA

Each sliding window of the MG-LDA model covered three sentences in all experiments. A document that contained S sentences would be represented by $S + 2$ windows. The distribution of words for each topic k was calculated according to the following equation:

$$\varphi_k^r(w) = \frac{n_w^{k,r} + \beta^r}{n_w^r + W\beta^r} \tag{1}$$

[6] https://github.com/bramrodenburg/nlp2015

[7] one Gibbs iteration with the MG-LDA model on the software product data with 20 global and 8 local topics would take on average two minutes and therefore one experiment could take between 16 to 20 hours

Where W is equal to the vocabulary size of the corpus and r denotes the global or local context. [8] $n_w^{k,r}$ denotes the number of times a word was assigned to a global or local topic k and n_w^r is the number of times a word is assigned to a global or local topic.

Pseudocode **Multi-Grain LDA**

Input: list of documents & sentences & words, number of global topics, number of local topics, number of Gibbs iterations

Output: word distributions over global and local topics, distributions of global and local topics

```
/* Initialize assignment of words to windows & topics                */
for each document d do
    for each sentence s_d in document d do
        for each word w_{d,i} in sentence s do
            w_{d,i}.window ← randomly choose a sliding window
            w_{d,i}.globalLocalContext ← randomly choose global or local topic indication
            if w_{d,i}.globalLocalContext == global then
                w_{d,i}.topic ← randomly choose a global topic
            else
                w_{d,i}.topic ← randomly choose a local topic
InitialzeCounters(w)

/* Gibbs sampling iterations                                         */
while iterations <= Number of Gibbs iterations do
    for each document d do
        for each sentence s_d in document d do
            for each word w_{d,i} in sentence s do
                DecreaseCounters(w_{d,i})
                /* build conditional distribution                    */
                for each sliding window v do
                    for each global topic k.gl do
                        P(v_{d,i}=v, r_{d,i} = gl, k_{d,i} = k.gl | v_{-d,i}, r_{-d,i}, k_{-d,i}, w) ←
                        CalculateConditionalProbability(d, s_d, w_{d,i}, v_{d,i}, k_{d,i})
                    for each local topic k.loc do
                        P(v_{d,i}=v, r_{d,i} = loc, k_{d,i} = k.loc | v_{-d,i}, r_{-d,i}, k_{-d,i}, w) ←
                        CalculateConditionalProbability(d, s_d, w_{d,i}, v_{d,i}, k_{d,i})
                /* sample new window and topic                       */
                w_{d,i}.window, w_{d,i}.globalLocalContext, w_{d,i}.topic ← P(v, r, k | w)
                IncreaseCounters(w_{d,i})
    iterations ← iterations + 1
/* build word & topic distributions                                  */
φ_gl, φ_loc, θ_gl, θ_loc ← BuildDistributions()
return φ_gl, φ_loc, θ_gl, θ_loc
```

Algorithm 2: Multi-Grain LDA

3.3 Results

The number of global topics (denoted as K^{gl}) was varied between 20 to 40 for LDA and MG-LDA. The number of local topics (denoted as K^{loc}) for MG-LDA was varied between 5 and 10. The results for the three different product categories are shown in table 2 on the following page and 3 on page 8 (for LDA resp. MG-LDA). An interpretation was given to the discovered topics where

[8] in the LDA model r is always equal to *global*

possible. [9] The manually assigned labels in the tables capture our perception of the topics. The last column of the tables shows the words that were most frequently sampled for that particular topic.

Table 2: Results of LDA

Datasets	Topics	Words
Software	Adobe Products	adobe, photoshop, cs2, illustrator, photo, image, photos, digital, editing, text
	Anti Virus	norton, mcafee, security, firewall, computer, symantec, spyware, virus, install
	Microsoft Office	office, microsoft, 2007, use, new, 2003, outlook, program, user, print, find, one
	Operating Systems	windows, vista, system, microsoft, mac, home, hardware, version, installed
	Quickbooks	quicken, time, money, program, business, quickbooks, need, online, questions
	Multimedia Software	software, nero, dvd, studio, works, easy, recommend, product, web, user, used
	Backups	drive, program, backup, disk, time, files, old, keep, every, going, back, got
	Product upgrades	version, new, features, upgrades, feature, better, previous, years, versions
	Customer support	support, software, product, problem, would, customer, help, service, call, tech
Sports and outdoor	Drinking bottles	water, bottle, top, bottles, drink, coffee, sigg, hot, plastic, size, cooler, sports
	Cycling	bike, ride, seat, riding, bikes, easy fit, front, mount, road, wheel, comfortable
	Gloves	gloves, good, use, great, bag, hands, hand, heavy, pair, grip, glove, wrist, blue
	Golf	ball, air, balls, swing, game, golf, play, clubs, case, hit, club, player, practice
	Presents	year, loves, son, kids, great, daughter, fun, love, little, christmas, toy, child, gift
	Price/Quality	great, price, product, good, quality, perfect, comfortable, money, highly, value
	Online shopping	product, amazon, shipping, service, received, box, order, arrived, customer
	Suunto sports watch	watch, compass, accurate, time, display, pedometer, features, suunto, altimeter
	Guns and rifles	gun, good, great, buy, scope, airsoft, fast, shot, clip, feet, shoot, even, best
	Knives	knife, blade, pocket, knives, small, handle, sharp, steel, tool, metal, edge
	Flashlights	light, batteries, battery, flashlight, night, lights, led, bright, power, see, hours
Magazines	Politics	bush, article, writers, political, tnr, war, new, liberal, economist, conservative
	News	journal, world, review, news, week, new, york, times, wall, street, editorial
	Fashion	magazine, fashion, vogue, patterns, knitting, pattern, photography, burda, good
	Software/digital	mac, reviews, macworld, computer, software, digital, ads, tips, photography, hardware
	Mens lifestyle	car, driver, road, fortune, news, world, articles, bike, business, policy, racing
	Cooking	recipes, cooking, food, cook, home, love, rachael, make, new, ingredients
	Home decoration	home, design, ideas, travel, articles, love, ads, decorating, beautiful, pictures
	Music	music, bands, reviews, rock, best, cover, articles, rolling, new, stone, spin, mojo
	Health & fitness	fitness, health, women, products, body, beauty, yoga, natural, men, life, advice
	Classical music	magazine, gramophone, new, music, recordings, classical, reviews, photos

4 Analysis

In order to assess the outcomes of the LDA and MG-LDA model we use a qualitative approach. The logic behind this is that it is hard to qualitatively determine the usefulness and correctness of topics.

As mentioned above the number of global topics was varied between 20 and 40 for both models. The quality of the global topics did not seem to increase when incrementing the number of global topics although the number of *usable* topics increased.
Titov and McDonald stated that the quality of the local topics was not influenced by the number of global topics as long as K^{gl} is twice the size of K^{loc}. In our experiments it seems that the quality of the local topics appeared to be slightly better when the ratio between the number of global and local topics was more skewed. [10]

[9] roughly half of the topics were omitted because we were not able to identify any clear meaning for them
[10] when K^{gl} was 3 to 4 times higher than K^{loc}

Table 3: Results of MG-LDA

Datasets	Level	Topics	Words
Sports and outdoor	Global	Body workout	exercise, muscles, workout, body, strength, resistance, wrist, machine, arms
		Drinking bottles	water, bottle, sigg, drink, sports, size, plastic, perfect, beautiful
		Sport watches	watch, compass, accurate, suunto, altimeter, time, barometer, functions, display, wrist
		Golf	swing, clubs, speed, golf, set, help, work, high, course
		Cycling	bike, ride, rack, like, seat, road, easy, wheel, tires, bicycle
		Yoga	mat, yoga, mats, bag, practice, like, good, great, floor, towel, feet, recommend
		Online shopping	back, amazon, product, service, send, order, company, shipping, never, customer, received, new
		Knives	knife, pocket, blade, opening, little, small, open, sharp, lock, nice, great, tools, army, box, carry
		Backpacks	bag, backpack, pack, bought, great, fit, back, size, perfect, around, good, big, love
		Assembly instructions	parts, screws, instructions, together, assembly, box, put, machine, back, piece, work, product, fit, plastic, directions, missing
		Flashlight	light, batteries, flashlight, lights, led, bright, time, night, minute, long , power, see, like, hours
		Gloves	gloves, bag, heavy, hands, pair, wrist, well, made, padding, straps, great, good, strap
		Guns and rifles	gun, good, scope, airsoft, like, great, shot, buy, shoot, accurate, clip, fps, rifle
	Local	Price/quality	good, price, product, worth, great, money, well, recommend, quality, like, purchase, overall, really, better, buy, happy, highly
		Presents	old, bought, year, loves, son, great, daughter, christmas, purchased, love, gift, kids, birthday, product
		Product assembly	easy, great, product, together, well, put, good, sturdy, assemble, shipping, works, love, assembly, instructions
		Ease & comfort	easy, use, comfortable, like, small, enough, seat, easily, well, great, bag, thing
Software	Global	Tax calculation software	tax, year, state, turbotax, return, taxcut, forms, taxes, data, years program, federal
		Language learning	spanish, language, learn, stone, rosetta, learning, would, words pimsleur, good, english, japanese
		Microsoft Office	office, word, 2007, microsoft, outlook, 2003, use, excel, features work, interface, email, powerpoint, professional
		Drawing & Design	adobe, illustrator, bible, comics, book, cs2, bridge, comic, spiderman, photoshop, coreldraw, curve
		Accounting software	program, quicken, business, quickbooks, like, product, version online, data, money, feature
		Games for children	game, play, games, fun, kids, chess, playing, child, children, nancy math, learning
		Windows Operating System	windows, vista, system, software, computer, upgrade, microsoft home, ram, drivers, operating
		Customer Support	support, software, customer, service, version, back, email, company download, hours, phone, trying
		Anti Virus	norton, internet, antivirus, security, system, firewall, virus, mcafee spyware, computer
		Route Planning	gps, route, roxio, program, streets, maps, trips, map, find, turn software, way, road
		Adobe Photoshop	adobe, photoshop, cs2, photos, file, images, use, tool, image, digital paint, control
		Backup software	drive, backup, hard, files, disk, would, computer, install, process backups, restore

In general it can be concluded that global topics capture product categories (e.g. different software categories) and local topic represent subtopics that can be associated with multiple product categories (e.g. Customer service or Quality/price). The topics extracted with the LDA model seem to capture less ratable feature as expected but still comprise ratable product properties. In addition we found that global and local topics of the MG-LDA model often contain *top words* that catch features of product categories e.g. *articles, ads, photography, photo, ideas, reviews or tips* for magazines and *upgrade*, *version* for software products. These are clearly ratable features appearing in both contexts.

The local topics of the *software* and sports & outdoor product reviews represent usable ratable product features (e.g. *price/quality, ease* & *comfort*). The local topics of *magazines* are in our opinion less usable as ratable topics, because the top words per topic are less coherent. As a result these labels are less trustworthy.

Table 4: Continued: Results of MG-LDA

Datasets	Level	Topics	Words
	Local	Price/quality	product, time, money, worth, great, recommend, good, easy, buy, really, worse, highly, like, much
		Customer support	support, product, problem, help, customer, software, service, tech, online technical, site, time, back, free
		Installation	software, money, installation, waste, new, could, home, installed, better
		Product upgrades	product, version, software, used, years, windows, mac, program, new, never
		User friendliness	use, easy, good, used, make, simple, learn, best, love, bought, dvd, yet, great
Magazines	Global	Politics	magazine, new, bush, years, people, article, issues, writers, political, tnr, war, liberal, economist, conservative
		News	journal, new, world, news, review, week, read, york, books, times, street, wall, london
		Fashion	magazine, fashion, vogue, patterns, knitting, look, people, articles, photography, burda, make, size
		Software/Digital	magazine, mac, articles, macworld, computer, software, digital, ads, tips, photography, hardware
		Men lifestyle	magazine, car, driver, road, fortune, cars, foreign, news, best, world, bike, business, racing
		Cooking	recipes, magazine, cooking, food, cook, recipe, home, rachael, easy, make, ingredients
		Home decoration	magazine, home, design, great, ideas, homes, decorating, love, beautiful, pictures
		Music	music, magazine, good, bands, reviews, rock, best, cover, rolling, stone, spin, great, mojo
		Health & fitness	magazine, fitness, health, women, products, body, beauty, yoga, great, natural, healthy, men, look, life, advice
		Travel	magazine, articles, people, places, living, find, travel, coast, reading, world, southern, coastal
	Local	Quality	magazine, great, articles, like, good, read, pictures, interesting, information, magazines, quality, well, stories
		Recommended	magazine, recommend, highly, like, new, best, great, articles, well, good, music, written, sports
		Price	magazine, subscription, issue, cover, read, worth, time, money, years, price, like, month, year, good
		Subscription service	magazine, subscription, issue, first, received, year, amazon, never, order, months, copy, service

Another observation is that the results of both models reveal that top words in global topics frequently contain words that express positive sentiments (e.g. *great, good, well, hightly, recommend*) and only sometimes negative sentiments. Obviously one could think that the reviews would mostly contain positive reviews but manual examination of the reviews shows that this is not the case. A possible explanation could be that the word ”not” is filtered out with the stopwords. One could also presume that negative reviews contain a higher variety of adjectives and adverbs than positive product reviews. Finally we surmised that it might be of influence that many of the negative reviews are very short compared to the positive reviews.

Certainly more research is necessary in order to clarify the mentioned conjectures.

5 Conclusion

In this paper we qualitatively compared the LDA and MG-LDA model on three product review datasets. The MG-LDA model extracted slightly better global topics than the LDA model and in addition captured reasonable ratable product features although the LDA results reveal that LDA can capture ratable aspects as well.

For future work a quantitative study next to a qualitative study could be performed in order to further compare the two models.

Acknowledgments

This paper follows from the course *Natural Language Processing 1* by Ivan Titov at the University of Amsterdam.

References

[1] D. Blei, A. Ng, and M. Jordan, *Latent Dirichlet Allocation*, Juornal of Machine Learning Research, vol. 3, 993-1022, 2003.

[2] J. Blitzer, M. Dredze, F. Pereira, Biographies, *Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification*. Association of Computational Linguistics (ACL), 2007.

[3] T. L. Griffiths and M. Steyvers. *Finding scientific topics*. Proc. of the Natural Academy of Sciences, 101 Suppl 1:52285235, 2004

[4] T. Minka and J. Lafferty. *Expectation propagation for the generative aspect model*. In UAI, 2002. https://research.microsoft.com/ minka/papers/aspect/minka-aspect.pdf.

[5] I. Titov, R. McDonald, *Modeling online reviews with Multi-Grain Topic Models*. WWW 2008 Beijing, 2008.