Customer Feedback Topic Modelling Using Online Latent Dirichlet Allocation

Yuhou Zhou

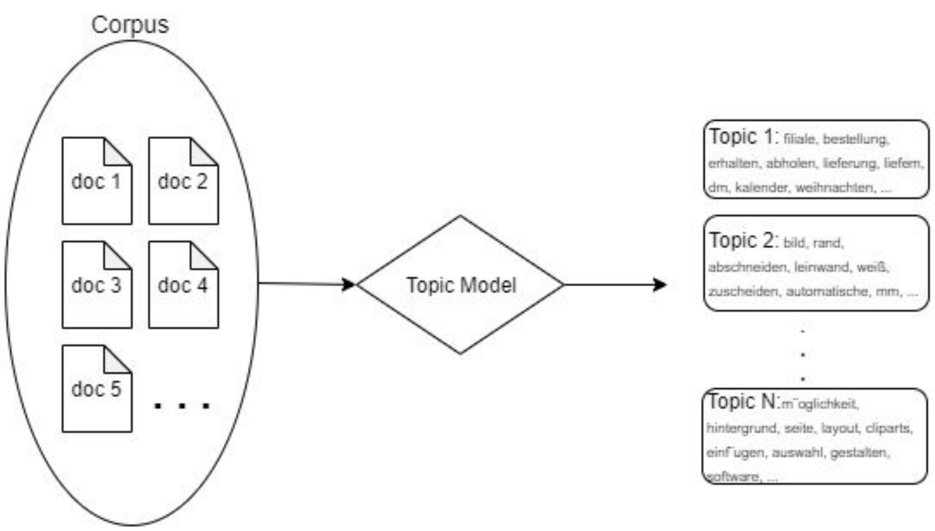
Outline

- Topic Modelling
- Online LDA
- Topic Coherence
- LDAvis
- System Architecture
- Implementation
- Experiment Result
- Outlook

Topic Modelling

In natural language processing, topic modelling is to discover abstract “topics” contained in a collection of documents.

Topic models includes Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), etc


Corpus

doc 1

doc 2

doc 3

doc 4

doc 5

. . .

Topic Model

Topic 1: filiale, bestellung, erhalten, abholen, lieferung, liefern, dm, kalender, weihnachten, ...

Topic 2: bild, rand, abschneiden, leinwand, weiß, zuscheiden, automatische, mm, ...

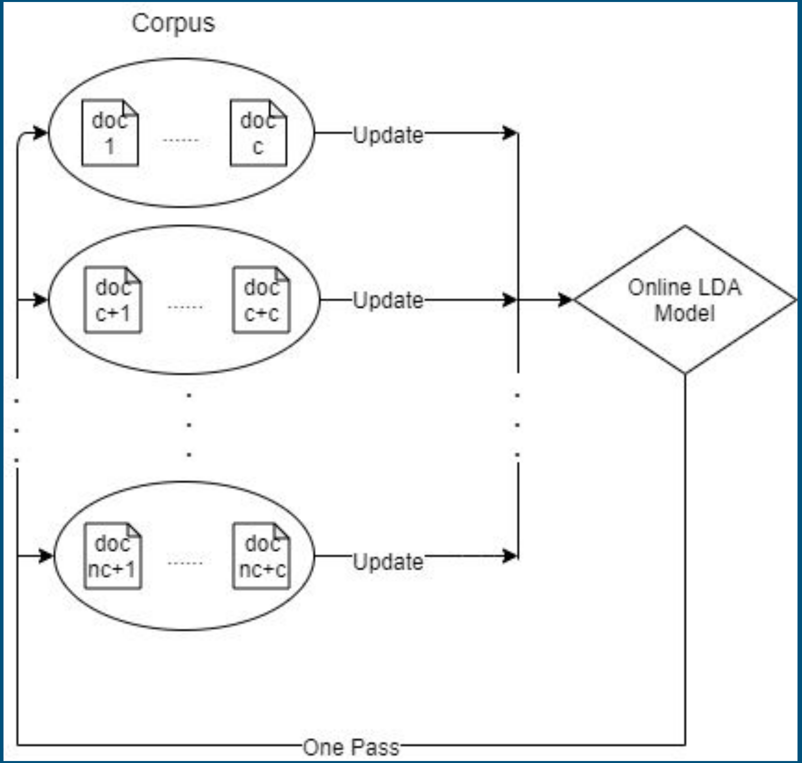
Topic N:m¨oglichkeit, hintergrund, seite, layout, cliparts, einf¨ugen, auswahl, gestalten, software, ...

Online LDA

Online Latent Dirichlet Allocation (Online LDA) is different from batch LDA in terms of model updating.

Batch LDA updates a model once per full pass of the corpus.

Online LDA updates a model many times per full pass of the corpus. In addition, it is faster, more accurate, and supports data from stream.



Topic Coherence

Model evaluation

There are many topic coherences, such as C_{UCI}, C_{UMass}, C_V, etc.

In our project, C_V is used because it is proved to have the best performance (Röder et al. 2015).

Higher the topic coherence, better the interpretability of the model.

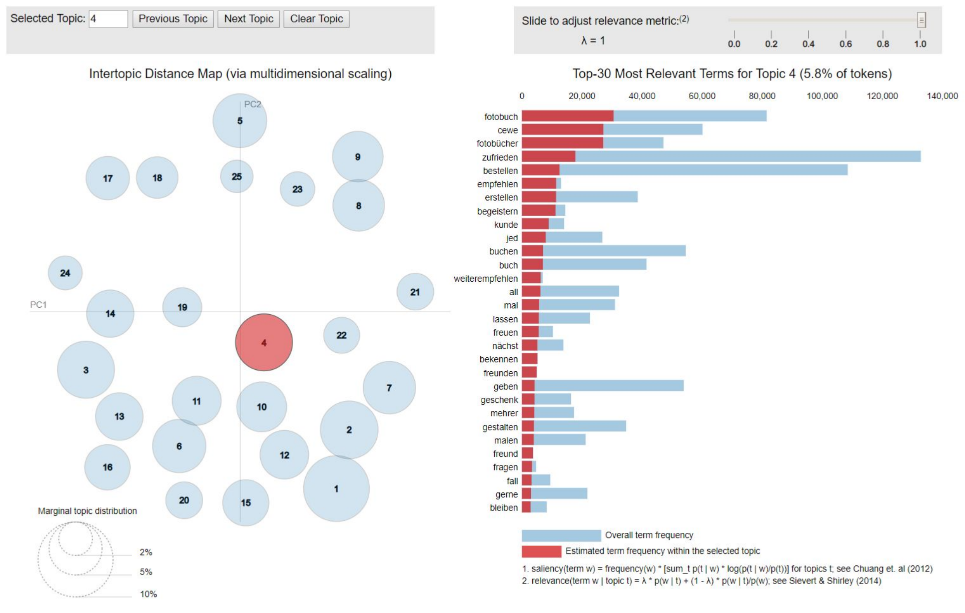
LDAvis

LDAvis is a web-based interactive visualization for LDA.

Rank the words in one topic by *relevance*, instead of by pure *probability*.

Change the value of λ, can change the setting of relevance.

When λ = 1 (default), the words are ranked by *probability*.



Selected Topic: 4

Slide to adjust relevance metric:(2)

λ = 1

Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Relevant Terms for Topic 4 (5.8% of tokens)

🟦 Overall term frequency

🟥 Estimated term frequency within the selected topic

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

System Architecture

Preprocessing, Modelling, Visualization

Preprocessing stage imports data from source and runs a standard preprocessing pipeline.

Modelling stage trains models and selects the best one by their topic coherence.

Visualization stage takes the best model and visualizes it.

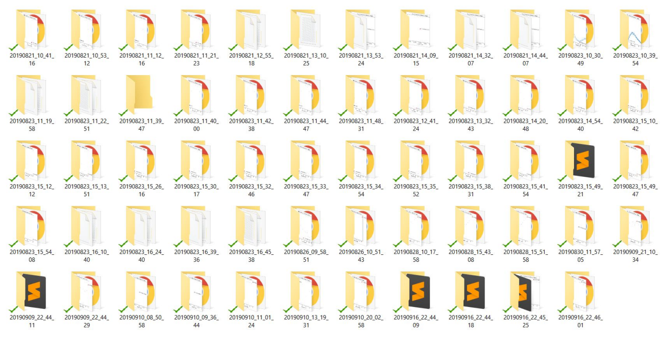
Implementation

Preprocessing: Apache Spark, spaCy, Pandas

Modelling: Gensim

Visualization: pyLDAvis, matplotlib

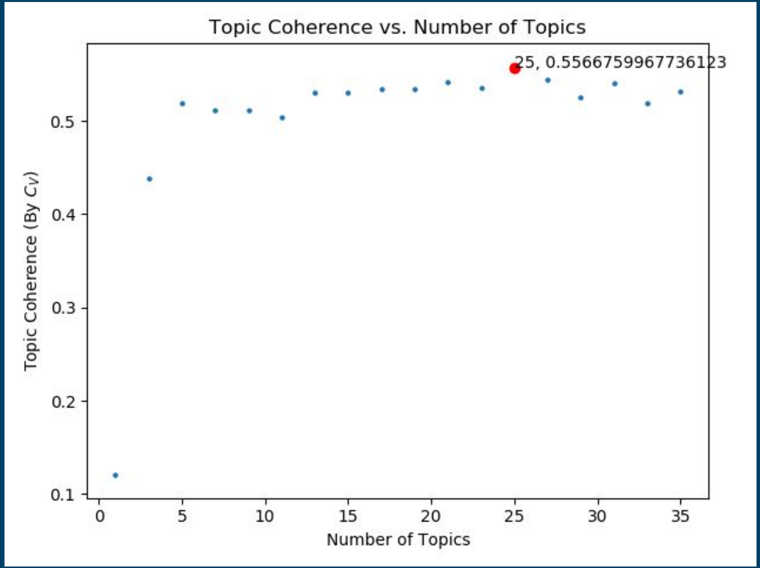
For every program run, all outputs are automatically archived, nice for reviewing.



Experiment Result

25 Topics

Experiment Result



Top 5 topics

Experiment Result

Topic 1: filiale, bestellung, erhalten, abholen, lieferung, liefern, dm, kalender, weihnachten, ...

Tipic 2: bild, rand, abschneiden, leinwand, weiß, zuschneiden, automatische, mm, kopf, dunkel, ...

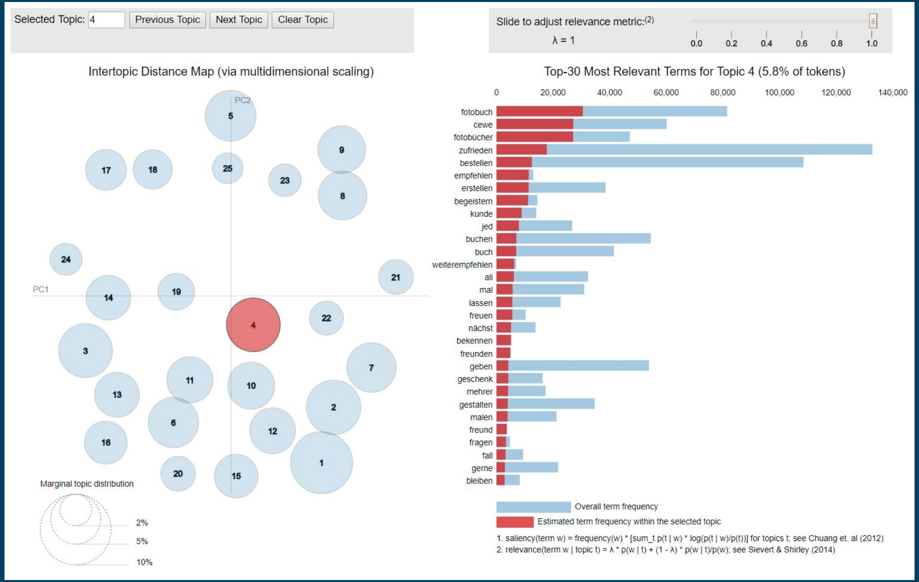
Topic 3: möglichkeit, hintergrund, seite, layout, cliparts, einfügen, auswahl, gestalten, software, ...

Topic 4: fotobuch, cewe, empfehlen, begeistern, kunde, weiterempfehlen, freund, zufrieden, ...

Topic 5: schnellen, lieferung, bearbeitung, zuverlässig, preiswert, zügig, prompt, einfach, unkomplizierte, unproblematisch, ...

Visualization

Experiment Result



Selected Topic: 4

Slide to adjust relevance metric:(2)

λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Outlook

- Generalizing the system to a broader use
- Improving evaluation metrics
- Developing a better graphical user interface
- Containerizing the system
- Using distributed Online LDA

Thank You
For Your Attention