Online Advertising In Website through Related Latent Topic Models Using Latent Dirichlet Allocation Algorithm

[1]M. Maheswari, [2]Sanjana Nahar, [3]Tanuj Agarwal
[1]Assistant Professor, [2,3]Student
SRM University, Chennai.

***Abstract* - In this paper, Search Engine Marketing (SEM) manages thousands of search keywords for their clients. The campaign management dashboards provided by interfaces to change search campaign attributes. Using these dashboards, users had created test variants for various bid choices, keyword ideas, and advertisement text options. And then, they used controlled experiments for selecting the best performing variants. Campaign management can easily become a burden on every advertiser. In order to target users in need of a particular service, advertisers have to determine the purchase intents or information needs of target users. Once the target intents are determined, advertisers can target those users with relevant search keywords. In order to formulate information needs and to scale campaign management with increasing number of keywords, we propose a framework called Topic Machine, where we learn the latent topics hidden in the available search terms reports. Our hypothesis is that these topics correspond to the set of information needs that best match-make a given client with users.**

***Index Terms* - Internet advertising, search engine marketing and Topic Modeling**

I. INTRODUCTION

The Internet has fostered on the ability to search the content people produce on the Web and find those pages that are highly relevant to a given query. Lucrative markets are created out of the information-seeking behavior of billions of people traversing the Web[1]. Eg. (DIY) social network creation platform that sells subscription based services. There are two options for the company to market their services: -

1) Launch a marketing campaign for displaying ads to the full set of Internet users.

2) Next, employ what is known as precision targeting as in “show my ad to any user who enters the query “create my own social networking site. If a user types into a web search engine “create my own social networking site”, then we may infer that this user wants to create a social network, and is potentially willing to pay for the provide service. There could be many such users browsing the Web. An advertiser can create an online advertisement that speaks exactly to the market segment containing these users or in other words create an advertisement to the specific purchase intent in question. This new type of advertising is attractive to both advertisers and customers at the same time. The query “create my own social networking site” is called a search keyword, or keyword for short, and the type of advertising that revolved around this notion is called keyword-based advertising. SEM agencies and practitioners manage thousands of search keywords on behalf of their clients. Any increase in the number of products being offered results in an increase in the number of search keywords being managed. Furthermore, as products tend to get more personal, catering to such niche segments increases the size of the keyword portfolio to manage dramatically. The campaign management dashboards provided by Google Ad words, Google Ad words Editor for bulk edits or Bing Ads have interfaces to change bid, scope, budget, and many other attributes per keyword or per groups of keywords. In addition, the management dashboards have features for advertisers to annotate their various bid choices, keyword ideas, and advertisement text options. Before creating a search campaign, an advertiser first needs to identify information needs (purchase intents), expressible as search keywords . These information needs are succinct descriptions of what each client product or service is offering to its users. For our example online platform to create white-label social networks, a mixture of information needs might look like “I want to create an online social community for fostering communication in our neighborhood and for increasing awareness to environmental issues concerning the welfare of our neighborhood. Furthermore, I want this online community to have private access”.

1.1 Motivation of the Paper

In order to match the users in need of a particular service with the client, which provides that service, advertisers have to determine the purchase intents or information needs of target users. Once the target intents are determined,
advertisers can target those users with relevant search keywords. In order to compile a relevant set of search keywords, advertisers analyze search terms reports, search query logs, and trend reports provided by ad-brokers. In
these reports, advertisers are exposed to how their target users express their hidden information need. Reviewing how users express their intent is very crucial as there is an impedance mismatch between how an advertiser describes an information need versus how a user expresses that need[4] . After reviewing reports, the advertiser may decide to add new search keywords to the

portfolio. As a consequence, the size of the keyword portfolio keeps increasing over time. The portfolio may be pruned by deleting or pausing keywords that perform poorly[5] , but the pruning has the risk of failing to capture the target information needs.

Given a large number of keywords, it is difficult to maintain keyword coherence within a campaign, and even more difficult to manage multiple campaigns consisting of a wide array of continuously evolving sets of keywords. Even though , management dashboards provide many features to slice and dice these keywords, they do not provide a semantic overlay or a topical structure on top of the existing campaigns. If such topical structures would exist, then an advertiser could easily manage these coherent topical structures rather than managing raw sets of keywords. This is critically important as more and more products and services get added to the product portfolio, the marketing of these offerings have to scale simultaneously. Hiring more people for the task is not a sustainable solution, and there are no guarantees for whether sales performance or return on investment (ROI) would also scale in line with such traditional marketing.

With a topical structure overlaid on top of the search campaigns, the advertisers can easily make campaign-wide changes, see the effects of such changes, understand the topical trends in the underlying search traffic over time, strategize into the future using the insights driven from structural trends, and target new market segments. All such management operations should be feasible while the underlying set of keywords scales to millions.

1.2 Contribution of the Paper

In order to scale SEM with an increasing number of product offerings while at the same time optimizing for conversions, we propose a framework called Topic Machine. In Topic Machine, we learn the latent topics hidden in the available search terms reports. Our hypothesis is that these topics correspond to the set of information needs that best match make the client with its users. For this purpose, we use a Latent Dirichlet Allocation (LDA) based topic model [6]. Since information needs may change over time or drift in concept, we learn dynamic topic models (DTM) by sequentially chaining model parameters in a Gaussian process across a well-defined epoch, e.g., weekly, bi-weekly, or monthly. In order to assess the quality of the models learned, we show the predictive power of the framework by measuring how well conversions per epoch can be predicted.

II. RELATED WORK

Probabilistic topic modeling is used to discover and annotate large archives of documents with thematic information[8] . With an increasing number of news, scientific articles, books, blogs, and web pages, it gets more difficult to categorize and search among the wealth of information. The idea behind LDA is to model documents as being generated from multiple topics (e.g., K topics) such that each of these topics is a probability distribution over a pre-defined vocabulary. Each document in the corpus exhibits topics in different proportions. In order to learn word distributions per topic and topic proportions per document, posterior probabilistic inference is used. With the observed documents in the corpus as output, hidden topical structure can be inferred by a deterministic variational method. In the variational inference procedure, a simpler distribution that contains free variational parameters is used to approximate the true posterior distribution. There are three sets of hidden variables each of which is governed by a different variational parameter: (1) a word distribution per topic, (2) a topic distribution per document, and (3) word-to-topic assignment per document[9] . All variables are assumed to be independent of each other. The variational parameters that maximize the log likelihood of the observations under the model are computed iteratively by continuous optimization using coordinate ascent.

The independence assumption in LDA has several drawbacks. For example, the presence of one latent topic may be correlated with the presence of another. Furthermore, since there is no notion of time in the model, documents in the corpus are exchangeable. However this assumption is inappropriate for many corpora including search terms reports. Search terms submitted by users reflect evolving content. In order to model time explicitly, dynamic topic models are introduced. In a DTM, each topic is dependent on its previous state and evolves in a state space model with Gaussian noise [7]. DTM has been successfully used to model thematic content change of scientific articles[10] . We use DTM over weekly search terms reports to learn an evolving conceptual model that can capture the hidden information needs and that can predict the sales performance better than alternative supervised approaches. LDA effectively provides a mechanism to reduce dimension within a given text corpus. With a K-topic LDA model learned over the corpus, each document is expressed as a K-dimensional feature vector of probability values. The sum of these values add up to one. As such, these vectors can readily be used as features for classification, for similarity

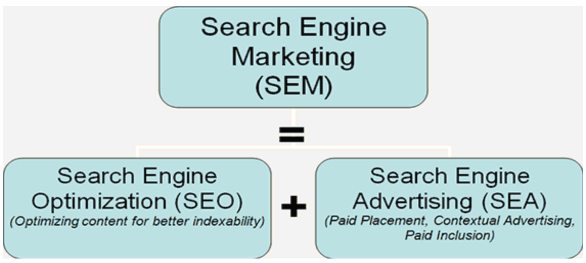

Fig: Overall Architecture

search as in similar topical proportions indicate similar documents, and for regression. Some textual data contain response variables such as movie reviews and their star ratings, comments and their total number of likes, and many other categorical responses. In such scenarios, the responses themselves can be modeled jointly with the associated documents in order to find the latent topics that best predict the response variable [11] . This model called supervised-LDA (sLDA) can be used for classification and regression tasks. The sLDA is used successfully to classify and annotate images [12] . In our settings, an sLDA model learned over weekly search terms reports and total weekly conversions for say the past 52 weeks can be used predict the future sales potential. This approach is implemented and considered as a competing technique to our proposed approach. Another supervised extension of LDA is to model each document with only a subset of the topics instead of using the full set of topics. The subset of topics are essentially dictated by human-annotation, metadata, or other automatic tags that are deducible from domain specific characteristics such as the semi-structure present in the text. Labelled LDA (L-LDA) [13] has been shown to outperform Support Vector Machines (SVM)[14] on certain text summarization tasks, e.g., representative snippet extraction per tag. Generally, LDA models are used on whole document corpora. In order to demonstrate the efficacy of inferring topic models on short text corpora, L-LDA has been applied to characterize tweets[15] . On two information consumption tasks as (1) recommending users to follow and (2) identifying interesting tweets, L-LDA performance has been shown to improve when it is coupled with TFIDF based information retrieval. This is because each of LDA and TF-IDF captures different aspects of textual similarity. Furthermore, L-LDA's classification and prediction performance is demonstrated on a large scale test conducted on 31:5M tweet data set[16] . L-LDA outperforms SVM on predicting top-k (k _ 3) topics that represent a user's profile and on predicting the similarity of two user profiles. The superior performance of LDA on classifying and regressing shorter texts supports our approach's fidelity for the application of LDA on search keywords.

When dealing with a large number of topics, it is beneficial to enforce a network structure on the topics learned. Higher order relations between topics make the topic exploration easier as well as managing these topics and reasoning about them. Relational Topic Model (RTM) uses the distance between topic proportions for linking documents and for super-imposing a network structure[17] on topics . In order to generalize the integration of metadata (as in L-LDA) and other contextual information about documents such as author, time of writing, and place of publication into the topic models learned, kernel topic models (KTMs) can be used to further enhance the trustworthiness of the latent topics discovered .

The management of search keywords benefit from model driven visualizations for text analysis and text summarization. For example, finding out descriptive key phrases for text visualization can be used to create thematic structure over the search keywords. Statistical and linguistic features are combined within a logistic regression model to learn phrase ranks . The top-N phrases are grouped using named entity recognition. In each group, a single phrase is selected to eliminate redundancy over the same entities. The final list of phrases are then displayed in a tag cloud. This work is complementary to our proposed approach as salient key phrases can be visualized in line with the topics discovered to communicate the textual content of the campaigns. Open Directory Project (ODP) ontology. Once the concept is found, the concept hierarchy within the ontology is used to find other concepts that are relevant to the primary concept. Each concept represents a set of web pages that are categorized under that concept within ODP. The webpage contents can then be used to suggest new keywords. To answer the question of how to best integrate a crowd sourced ontology into unsupervised probabilistic modeling requires further research. In similar vein to LDA, a generative translation model and a corresponding language model are used in SEM research to suggest search phrases for bidding. The translation model formulates the probability of a given search term to align with a corresponding term in the target webpage. The higher is the alignment probability, the more relevant is the search term to the target webpage. The translation probabilities are learned iteratively using the

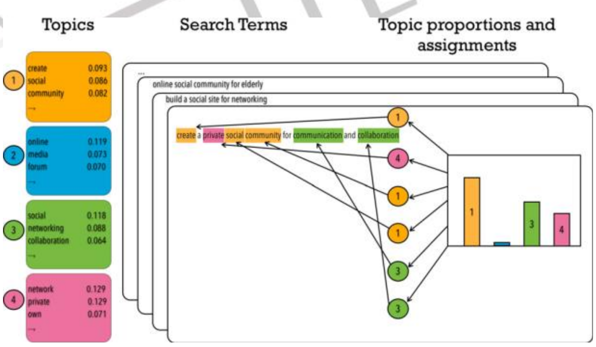

Fig 1: The intuitions behind Latent Dirichlet Allocation. We assume that some number of "topics", which are distributions over words, exist for the whole search terms collection (far left). Each search term is assumed to be generated as follows: First choose a distribution over the topics (the histogram at far right); then for each word, choose a topic assignment (the colored circles) and choose the word from the corresponding topic. The topics and topic assignments are illustrative.

Expectation Maximization (EM) algorithm. LDA has been used to estimate the query advertisability in sponsored search. A predictive model is built using words found in queries, where the contribution of each word to advertisability is the fraction of times the word is present in queries that get user clicks versus queries that do not get user clicks . LDA topical membership information is used as a binary feature for representing queries in a regression model. Contrary to our holistic approach of using

LDA for representing all search terms of a given campaign per epoch, LDA is used for enriching feature extraction from individual queries.

III. FOUNDATIONS AND METHODOLOGY

In this , we describe the basic ideas behind LDA[6] as it forms the building block in our framework. The intuition behind LDA is that search terms encompass multiple information needs. A search term also known as a search keyword is what a web user types in for querying a search engine. An advertisement keyword is what an advertiser uses for targeting search engine users. Keywords consist of tokens, each of which is a word of a natural language. For example, consider the search term “create a private social community for communication and collaboration” in Fig. 1. In this case, the user would like to create a social community, which should be private, and furthermore wants it to be used for enhancing community communication and collaboration. The words about creating online social communities, such as “create”, “social”, and “community” are highlighted in yellow and marked with number 1 to indicate topic 1; words about privacy in online social networks, such as “private”, are highlighted in pink and marked with number 4 to indicate topic 4; and words about social networking
sites for collaboration, such as “communication” Fig. 1. The intuitions behind Latent Dirichlet Allocation. We assume that some number of “topics”, which are distributions over words, exist for the whole search terms collection (far left). Each search term is assumed to be generated as follows: First choose a distribution over the topics (the histogram at far right); then for each word, choose a topic assignment (the colored circles) and choose the word from the corresponding topic. The topics and topic assignments are illustrative. and “collaboration”, are highlighted in green and marked with number 3 to indicate topic 3. With this insight, we can see that this search term blends private networks, social
communities, and networking sites. LDA assumes that there is a generative process that gave rise to the observed search terms. And the probabilistic inference is used to find out the hidden parameters that characterize the generation. A topic is a multinomial distribution over a chosen vocabulary of words found in the search terms. For example, networking sites topic has words about social networking with high probability, while private networks topic has words about privacy issues with high probability. The assumption here is that these topics have been specified before the data arose. With the topics specified, the algorithm for generating a search term is given in Algorithm 1. Since the algorithm itself is self explanatory, we omit the textual description. The goal of LDA is to find out the topic structure given the observed search terms. The topic structure consists of the topics themselves, per-document topic distributions, and per document per word topic assignments. The topic structure is the hidden structure. The hidden structure is more formally described as follows:

1) Let K denote the number of topics, and N denote the number of words in the dth search term.
2) Topics are b_1:K, where each b_k is a distribution over words in the vocabulary (see word distributions on the far left of Fig. 1).
3) The topic proportion distribution for the dth search term is u_d, where u_d ;k is the
frequency of topic k in search term d (see the histogram on the far right of Fig. 1).
4) The topic assignments for the words in the dth search term are z_d, where z_d;n is the topic assignment for the nth word in the search term (see the coloured and numbered circles in Fig. 1).
5) The observed words for search term d are w_d, where
w_d; n denotes the nth word in d.

```
Input: Topics (there are K topics)
Output: topic assignments to words, topic proportions
N ← length of the search term.
topicProportions ← {topic1:0, topic2:0, ..., topicK:0}.
topicAssignments ← {1: None, 2: None, ..., N: None}.
topicDist ← randomly choose a distribution over
  Topics.
for i in range(N) do
   topic ← sample a topic from topicDist.
   wordDist ← retrieve word distribution for topic.
   word ← randomly choose a word from wordDist.
   topicProportions[topic] ← topicProportions[topic]
   + 1
   topicAssignments[i] ← topic
end for
return topicProportions, topicAssignments.
```

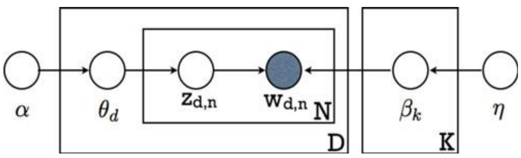

Fig: 2 Graphical Model of LDA. Each node is a random variable. The hidden nodes,per term, topic assignment and topics themselves are shown as unshaded part. The obseverd words for search are shown as shaded part. The rectangles represent the replication in plate notation.

3.1 Topic Machine

Given search terms $D_1;D_2; \ldots;D_T$, a T-dimensional conversions vector y, the chosen number of topics K, LDA hyperparameters a and h, , and a selective method to use for choosing the vocabulary V , a Topic Machine can be built shown in Algorithm 2. First, the vocabulary V is determine on the union of all search terms at step 1. In between steps 2- 6, each search term collection D_t is formatted according to LDA-C format using V to construct input observations for the model. Using these observations at steps 7 and 8, a DTM model with K topics is learned for the input hyper parameter values of a and h. At step 9, features that represent topical density per epoch are extracted from the learned machine. Using X and conversion values input y, a correlation coefficient based topical trends analyzer are learned at steps 10 and 11 respectively. The output at step 12 is the complete TopicMachine.

Input: Search terms $D_1, D_2, \ldots, D_T$, $\mathbf{y}$, K, α, η, Selector
Output: TopicMachine
1: $V \leftarrow$ ChooseVocabulary($\bigcup_t D_t$, Selector)
2: Observations $\leftarrow$ []
3: **for** t in range(T) **do**
4: o $\leftarrow$ FormatLDA-C(D_t, V)
5: Observations $\leftarrow$ Observations + o
6: **end for**
7: β, θ, $z \leftarrow$ LearnDTM(Observations, K, α, η)
8: TopicMachine[Model] $\leftarrow \beta$, θ, z
9: $\mathbf{X} \leftarrow$ ExtractFeatures(TopicMachine)
10: TopicMachine[Regressor] $\leftarrow$ TrainLasso($\mathbf{X}, \mathbf{y}$)
11: TopicMachine[Trends] $\leftarrow$ ComputeTrends($\mathbf{X}, \mathbf{y}$)
12: **return** TopicMachine

IV. FUTURE ENHANCEMENT

In Future, The insights driven from structural trends, and target new market segments. All such management operations should be feasible while the underlying set of keywords scales to millions. The performance of LDA-based models depends on the choice of the initial starting point. Some model fits may get stuck in local optima. Therefore, DTM-based approaches pose a great advantage in terms of evolving from an already established and plausible model fit. First, an sLDA model with a superior predictive power is found after repeated iterations. After that, a DTM-based model may evolve from it thereafter. SEM Advertisers repeatedly review search terms reports to identify relevant or non-relevant search terms. Relevant search terms are used to expand existing keyword campaigns while non-relevant search terms are used to reduce the size of these campaigns. This problem is a classification problem on the search term space. An adaptation of sLDA with supervision in terms of relevance can be used to build a topic-based classifier for asserting relevance judgements automatically.

V. CONCLUSION

We proposed a framework called TopicMachine where we learn the latent topics hidden in the available SEM search terms data. Our foundational hypothesis is that there are a set of information needs hidden in and behind the search terms as a collection; the latent topics that are discovered through probabilistic inference over this collection correspond to these information needs. The topical structure stands as a viable tool to manage SEM campaigns with precise targeting of users in terms of relevance and to optimize for conversions. TopicMachine uses an LDA-based topic model. Since information needs may change over time or drift in concept, we learn dynamic topic models by sequentially chaining model parameters in a gaussian process across a welldefined epoch. We assessed the quality of the models learned in TopicMachine by showing the predictive power of the framework. We measured how well the conversions can be predicted with features extracted from TopicMachine model.

KEYWORDS

Search Engine Marketing-Search marketing is the process of gaining traffic and visibility from search engines through both paid and unpaid efforts. Originally called “search engine marketing,” the shorter phrase “search marketing” is now often used as the umbrella term over SEO and SEM. Search engine traffic is a non-intrusive method of Internet marketing

Internet Advertising-Online advertising, also called Internet advertising, is a form of advertising which uses the Internet to deliver promotional marketing messages to consumers. Online advertising is a large business and is growing rapidly.

Topic Modeling- Topic modeling is a form of text mining, a way of identifying patterns in a corpus. You take your corpus and run it through a tool which groups words across the corpus into 'topics'.

ACKNOWLEDGEMENTS

The author would like to thank Grou.ps Inc. for providing the campaign data set, Sean Gerrish from Sift Science for helping out with DTM compilation, Kai Chang for the implementation of parallel coordinates on D3, and Chong Wang from Carnegie Mellon University and David M. Blei from Princeton University for making the LDA implementations publicly available.

REFERENCES

[1]. D. Easley and J. Kleinberg, Networks, Crowds, and Markets: Reasoning about a Highly Connected World. New York, NY, USA: Cambridge Univ. Press, 2010.

[2]. R. Kohavi, R. Longbotham, D. Sommerfield, and R. M. Henne, "Controlled experiments on the web: Survey and practical guide," Data Mining Knowl. Discovery, vol. 18, no. 1, pp. 140–181, 2009.

[3]. C. D. Manning, P. Raghavan, and H. Schutze, Introduction to Information Retrieval. New York, NY, USA: Cambridge Univ. Press, 2008.

[4]. B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. S. de Moura, "Impedance coupling in content-targeted advertising," in Proc. 28th ACM Int. Conf. Res. Develop. Inform. Retrieval, Aug. 2005, pp. 496–503.

[5]. P. Rusmevichientong and D. P. Williamson, "An adaptive algorithm for selecting profitable keywords for search-based advertising services," in Proc. 7th ACM Conf. Electron. Commerce, Jun. 2006,pp. 260–269.

[6]. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," J. Mach. Learning Res., vol. 3, pp. 993–1022, Mar. 2003.

[7]. D. M. Blei and J. D. Lafferty, "Dynamic topic models," in Proc. 23rd ACM Int. Conf. Mach. Learning, 2006, pp. 113–120.

[8]. D. M. Blei, "Probabilistic topic models," Commun. ACM, vol. 55, no. 4, pp. 77–84, 2012.

[9]. D. M.w Blei and J. D. Lafferty, "Topic models," in Text Mining: Classification, Clustering, and Applications, London, U.K.: Chapman & Hall/CRC Press, 2009 pp. 71–94.

[10]. S. Gerrish and D. M. Blei, "A language-based approach to measuring scholarly impact," in Proc. 27th Int. Conf. Mach. Learning, 2010,pp. 375–382.

[11]. D. M. Blei and J. McAuliffe, "Supervised topic models," in Proc. Advances Neural Inform. Process. Syst. 20, 2008, pp. 121–128.

[12]. C. Wang, D. Blei, and F.-F. Li, "Simultaneous image classification and annotation," in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2009, pp. 1903–1910.

[13]. D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: A supervised topic model for credit attribution in multilabeled corpora," in Proc. Conf. Empirical Methods Natural Language Process., 2009, pp. 248–256.

[14]. C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learning, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[15]. D. Ramage, S. T. Dumais, and D. J. Liebling, "Characterizing microblogs with topic models," in Proc. Int. Conf. Weblogs SocialMedia, 2010, pp. 130–137.

[16]. D. Quercia, H. Askham, and J. Crowcroft, "TweetLDA: Supervised topic classification and link prediction in twitter," in Proc. 3rd Annu. ACM Web Sci. Conf., 2012, pp. 247–250.

[17]. J. Chang and D. M. Blei, "Hierarchical relational models for document networks," The Ann. Appl. Statist., vol. 4, no. 1, pp. 124–150, 2010.