LongTale

A Recommendation System for E-Commerce, Social Networks, Online Content Providers, and Brick&Mortar Retailers

Abstract

LongTale is a computational platform for generating personalized recommendations. Consumers today are exposed to a bewildering array of products and services. Delivering a personalized marketing message from among a myriad amout of possibilities is hard, and carries the risks of being irrelevant and intrusive. Personalization means aligning the marketing message with the individual tastes and preferences of consumers. This will save time, save money, better inform the customer, and match the needs of consumers with the relevant value propositions. Recommendation systems achieve this by guiding the consumer to a small subset of product/service space that she might be interested in. In essence, a recommendation system is a match-maker: it matches the content of a product or service with the specic tastes and preferences of an individual consumers, and produces

suggestions that might help to solve "the complexity of choice" problem faced by consumers. A recommendation system is, *au fond*, the primary marketing tool in an online setting where the sole delivery means of a marketing message is through **personalized recommendations** given in an electronic medium. The need for a recommendation system is proportional to the number of active customers, and more so to the number of products&services offered. Hence a recommendation system is more useful for e-retailers, social networks, online content providers, and brick&mortar retailers where consumers are exposed to an overwhelming number of choices she can potentially make.

There are two basic approaches to building a recommendation system: **Content based** systems use consumer demographics, and product characteristics data to match consumers with products. **Collaborative Filtering** uses transactions data to produce its recommendations. LongTale is a hybrid system that can utilize both demographics and transactions data to produce superior recommendations for each consumer. However, one should be reminded that transactions data carries bulk of the information content useful in producing relevant recommendations. LongTale uses statistical and machine learning algorithms to build non-linear maps between demographics/transactions data and consumer preferences for products&services. These non-linear maps are used to produce personalized recommendations for each consumer.

LongTale uses an ensemble-learning approach to build its recommendations: Depending on the scale and complexity of data, LongTale uses different statistical and machine learning algorithms to build several models, and eventually combines the results of these algorithms to build a final ensemble model. Though the recommendations produced by different algorithms might be correlated, each has unique information content as well. The ensemble approach is a tried-and-tested approach producing far superior recommendations over those produced by single models.

The computational complexity of LongTale algorithms might be huge: The implicit data by a mid-sized e-commerce company might easily scale to the order of billions of transactions per year. Analyzing this data and attaching a recommendation score to each potential consumer-product pair is momentous. The ensemble approach makes the case more difficult by the necessity of running different algorithms on the same data. Capturing temporal dependency of recommendations (seasonality, non-stationarity) is another factor increasing this complexity. LongTale employs parallel computing in order to handle this computational complexity. LongTale is designed to run on shared memory and distributed memory architectures, and exploit the benefits of modern day multi-CPU, multi-core hardware.

LongTale is NOT a plug-and-play software: A specific LongTale project maps to building thousands of statistical models that must be initially tuned by analysts to fit the peculiarities of data, and cater to the particular needs of the client. LongTale was designed to serve as Software-As-A-Managed Service. In this framework, client provides the data on a regular basis, and LongTale analysts provides recommendations in return. The pricing formulae of LongTale services is value-based: It is based on conversion of recommendations into actual purchases, and additional revenue contribution by recommendations. This utilitarian pricing scheme was selected since it is a win-win proposition for both parties: Client does pay only for the added benefit from the services, and vendor's pay is not limited to a one-time-only-fee.

1 LongTale Features

Digitization of marketing media is a boon and a bane at the same time for the marketeers: There are simply more offers to market to consumers having no spatial or temporal boundaries. However, the sheer scale of product-gamut, customer population, and transaction sizes render classical advertising and marketing campaigns less effective. When the scale enters the marketing game, **personalization** is indispensable and unavoidable. Making the right offers to the right customer becomes more challenging in mediums where interaction with customers is purely digital, products are numerous, and sometimes customers are anonymous. The response to this challenge is managing this complexity with **personalization technologies**.

Personalization can transform this scale challenge into a marketing machine: It presents the opportunity to introduce millions of products to millions of customers, an effort previously not possible with classical marketing strategies. Theoretically, one can realize personalization by collecting demographics, pscyhographics, and financial data from consumers. However, collecting and maintaining these data from consumers are notoriously difficult, and rising privacy concerns prohibit this further. Meanwhile, formulation and maintenance of possibly thousands of marketing strategies in the form of business rules is both sub-optimal, costly, and borders on being impossible. However, transactional (behavioral) data is immensely rich in order to understand customer needs, and achieving the personalization in the proper way.

LongTale is a recommendation system that can provide the personalization through massive analysis of transactional data, context data, and content data to automatically formulate targeted offers from among millions of possibilites towards millions of consumers. An offer might be a product in an e-commerce web site, a product in a bricks&mortar store, a digital ad in a web-site, a news item in an electronic paper, a

video content in a social network, or any item worth offering in a choice-set. Hence, LongTale allows a marketeer to target the right offer to the right customer at the right time through the right channel with the right message. The necessary information is stored in the vast transactional information stores of the clients. They just need to be transformed in the right way.

LongTale is a highly accurate, computationally efficient recommendation system that is capable of producing recommendations in offline and real-time scenarios. Following features empower LongTale:

- LongTale has a spectrum of algorithms that learn consumer preferences with different learning paradigms. A single LongTale recommendation is an ensemble-aggregate of several LongTale algorithms.

- LongTale algorithms account for **consumer biases**: Though collaborative filtering plays the center stage in producing recommendations, they need to be adjusted for consumer biases: Handling repetitive purchases in a single category is such a bias.

- LongTale algorithms account for **item biases**: An item might be popular temporarily, or it might be a mainstream item. LongTale algorithms account for those biases.

- LongTale algorithms account for **temporal biases**: Consumer profile might change; consumer preferences might vary seasonally; item popularities are dependent on time (there might be peaks, cycles, trends, etc.). One of the glaring deficiencies of most recommendation engines is that they could not account for these temporal dependencies. LongTale can.

- Heterogeneous consumer profiles: Household effect, geographic displacement, eclectic tastes confuse recommendation algorithms as to produce biased recommendation

towards dominant sub-profile. LongTale partitions consumer profiles, and produces recommendations for each latent profile.

- LongTale algorithms account for **context**: Consumers may show new preferences that are in contrast with their established historical profiles. A recommendation system should be flexible enough to adapt to new context; regard the new context as the clue and historical profile as the backdrop to produce context-relevant recommendations. LongTale follows this principle for producing recommendations in real-time.

- Products obey a hierarchy, and/or they might have features. LongTale produces recommendations at atomic product level, at all levels of a product hierarchy, and at the level of product features. Hence, LongTale might suggest you Brothers Karamazov, as well as other Dostoyevski novels, or a crime novel at the same time.

- LongTale accounts for **Cold Start** problem for consumers and products. New consumers, and new products present difficulties for recommendation systems. As LongTale algorithms produce recommendations at different levels of a product hierarchy, they are able to generate recommendations for new items. Meanwhile, LongTale might employ similar user profiles which might be used to generate recommendations for new users.

- LongTale is able to produce recommendations for sparse-profiles: Similar to Cold-Start problem.

- LongTale uses both explicit data and implicit data to produce its recommendations. **Implicit data** refers to page-views, purchases, shopping cart contents, wish-list contents, searches, etc. **Explicit data** refers to ratings, and product evaluations.

LongTale properly adjusts the weights of each of the different data types above to produce its accurate recommendations.

- LongTale generates a **Recommendation Mix** for each consumer: This recommendation mix consists of an array of different types of recommendations: Cross-sell offers, up-sell offers, More-like-this offers, rare item offers, new item offers, recent item offers, recently popular offers, category level offers, feature-level offers, context-sensitive offers, etc. constitute the recommendation mix for a consumer. This mix is then presented to consumer in different contextual settings. For example, the type of recommendations shown in shopping-cart page is different from the recommendations shown in the main page.

- LongTale allows incorporating filters in terms of business rules in composing the recommendation mix.

- LongTale is able to adjust its recommendations to optimize different objectives (order size, total sales amount, total number of purchases, etc.)

- LongTale operates in two modes: **Offline**, and **Real-time**. Offline recommendations are built prior to interaction with the consumer. Real-time recommendations are produced during interaction with the consumer. LongTale can process click-stream data to generate recommendations in real-time.

- LongTale can produce canned reports about site-traffic as well as recommendation performance. Depending on client requirements, LongTale is flexible enough to build and deploy new reports.

- LongTale algorithms are computationally efficient: LongTale employs best-of-the-breed optimized code, and massive hardware parallelism to meet stringent perfor-

mance criteria.

- LongTale is **SaaMS**. As such, deployment of new functionality is fast, and client does not need to allocate any resource for building the system.

- LongTale can produce recommendations for fast-moving-content such as news, and ads.

- LongTale can produce recommendations for items that are defined by unstructured metadata. For example, in order to recommend a news item that has just arrived, it has to be classified under a certain category. LongTale employs semantic indexing to achieve this classification. After semantic indexing, recommendation engine handles the rest.

2 LongTale Technology

A recommendation system should be *accurate* in producing relevant recommendations, and it should be *scalable* to handle billions of transactions that belong to millions of consumers. This section gives a brief summary of how LongTale achieves its accuracy and scalability:

2.1 Algorithms

LongTale employs a family of machine learning and statistical algorithms to produce accurate, high performance recommendations in real-time. The philosophy of LongTale in building a recommendation system is to tailor the system into client's requirements and constraints. The challenge in this process is to optimize the trade-off between accuracy (requirement), and performance (a constraint). Once the optimal point is determined

after discussions with the client, LongTale picks the appropriate algorithms to build a recommendation system based on **ensemble modelling**.

An ensemble model is an **ensemble** of models that are collectively superior to its individual constituent models. Each model in an ensemble model is the result of application of a specific algorithm to data. Machine learning and statistical algorithms "learn" the data in specific ways. An ensemble model combines the knowledge produced by each algorithm into a superior one. Hence, the selection of algorithms that LongTale will use in a recommendation system is entirely dictated by client requirements and constraints. If the client desires the best recommendation system money can buy, LongTale will employ ensemble modelling with several algorithms to produce the best models.

LongTale can currently employ following algorithms and methodologies to build accurate and efficient recommendation systems:

- Item-to-item collaborative filtering: Probably the simplest and the most widely used recommendation algorithm. Item-to-item CF algorithms build similarities between recommendable propositions, and weights these similarities according to past user preferences to generate new recommendations.

- User based collaborative filtering: A relative of item-to-item collaborative filtering algorithms. It employs similarities between users (computation of similarities between users is not trivial) to come up with peer-group recommendations.

- Generalized Additive Modelling with Regularization: One of the best algorithms that provide good balance between accuracy and computational performance. GAMR is an extension of generalized linear modelling in two important directions:

    1 Capturing the possible non-linear effects between inputs and output(s).

2 Regularizing the solution to prevent overfitting, thus producing robust models.

- Time Dependent Regularized Matrix Factorization : The second work-horse of the family of LongTale recommendation algorithms. In essence, matrix factorization extracts **latent user factors**, and **latent item factors** by *factoring* transactions data. User factors capture the user proclivities towards items, e.g. an entry in a user factor might show the high tendency of user to buy "luxury" items. Item factors capture the latent item factors that code the hidden item clusters, e.g. an entry in an item factor might indicate that item *is* a luxury item. User factors, and item factors collectively identify the interest of a user in an item. LongTale adds the time dimension to matrix factorization in order to account for changing preferences in time.

- Boosting Algorithms: Boosting algorithms are meta algorithms that use other machine learning or statistical learning algorithms to build a better model than the individual models produced by each algorithm. Boosting has emerged as a powerful learning methodology along with support vector machines (SVM) in the last decade. SVMs are very costly to be used as scalable recommendation algorithms, and thus currently is out of option.

- Ensemble Learning: Ensemble learning combines many models to produce a best model whose accuracy surpasses the accuracy of its individual constituents. Each learning algorithm puts forward a different specification (a "formula") to learn the regularities buried in data. Ensemble learning, in a way, optimally combines the "opinions" of each model about a case (e.g., best recommendation set to a customer) into an "overall opinion" that is collectively better than any individual opinion. LongTale routinely employs ensemble modelling to merge the outputs of

several algorithms into a "best model".

2.2 Computations

LongTale computations are CPU-intensive and use large amounts of physical memory. **Scalability** of the system is ensured by exploiting following modern day technologies:

- **Shared Memory Parallelism:** LongTale algorithms can exploit the modern day multi-core arhitecture of CPUs (Intel is currently testing 48-core CPUs that will be released in a couple of years. This paradigm shift is revolutionizing the code parallelism). By using optimized multi-threading, LongTale implements shared-memory parallelism in performance-conscious parts of the code.

- **Distributed Memory Parallelism:** Most of the parallelism jobs in LongTale system are *embarrassingly simple*, i.e. parallelism in a distributed computing architecture is provided by partitioning the input data, let different machines process the data assigned to them, and gather the results thus produced. This scheme breaks down when the machines need to pass messages to complete jobs (i.e., one machine's CPU needs to read/write to other machines' memory). LongTale uses message passing mechanism to implement distributed memory parallelism.

- **In-Memory Analytics:** LongTale has to respond fast to possibly thousands of recommendation requests per second in a real-time scenario. LongTale employs fast in-memory computations instead of expensive database calls to handle these requests.

- **High performance libraries:** LongTale uses the best-in-class Fortran and C numerical libraries to meet the high performance computing requirements of a scalable recommendation system.

3 LongTale Application Areas

In this section, some sample applications of LongTale recommendation system are provided. They are provided for illustrative purposes only: They are neither complete nor exhaustive:

3.1 Bricks&Mortar retailers

In a nutshell, LongTale produces a set of offers for each consumer at a certain point in time. Hence, LongTale applications at a physical store are varied depending on the time, the context, and the physical location of the delivery of this set of offers. A set of efficacious applications stand out:

a. Coupon printing at kiosks: Before customer does any transaction, she can use her identification card (loyalty card, store card) at an in-store kiosk for printing the set of offers that is redeemable at the current shopping session.

b. In-store SMS marketing: If customer's presence in the store can be identified by the client, it can send an SMS message to the consumers mobile to inform her of the set of offers that can be redeemed at the current shopping session.

c. Coupon printing at the cashier: After customer has finished her current shopping session at the cashier, a new set of relevant offers based on current shopping basket can be printed in order to maximize the relevance of offers. This new coupon can be redeemed in the next shopping session.

d. Digital coupon printing: If the client has a web presence (not necessarily an e-commerce site), it can communicate personalized offers to its clients on the web: The consumer enters the web-site; identifies herself; inspects the personalized offers

for herself; prints the offers in the form of a coupon, and redeems the coupon at the next shopping session at the physical store.

e. Offline E-mail/SMS marketing: A routine marketing application of bricks&mortar retailers is E-mail/SMS marketing. However, these applications suffer from the lack of relevance. The entire promise of LongTale is *creating relevance*. Hence, one immediate application of LongTale is to perfect E-mail/SMS marketing initiative of the retailer.

3.2 E-commerce

The personalization provided by recommendation systems is the-bread-and-the-butter of marketing effort in e-commerce realm. In fact, marketing strategy based on a recommendation system is *the* marketing method in e-commerce. The outstanding champions in e-commerce arena (**Amazon, E-Bay, NetFlix**) employ recommendation systems to generate their offers. LongTale determines what to offer, when to offer, and how to offer depending on the context of the interaction, and the history of client transactions. A recommendation mix is presented to the client before she makes any page-views (offline recommendation computations), and the set of offers are fine-tuned as she surfs over the product web pages (real-time recommendation computations). The goals are the same: To present relevant offers in order to attract consumer interest with the final ends of increasing stickiness, increasing click-through rates, increasing conversion rates, and increasing transaction sizes.

3.3 Content providers

The revenue source of content providers (apart from possible subscription fees) is digital advertising. The marketing problem is then to determine what to advertise to each customer. The solution to this problem is to determine the target population for each ad via a recommendation system. Meanwhile, customer should spend as much of their time as possible in the site(stickiness) to see more relevant ads. In order to increase stickiness, the consumers should be presented with relevant content as it arrives. Hence, a recommendation system performs two critical functions in order to increase revenue:

1 To increase stickiness by continuous presentation of relevant content as determined by a recommendation system,

2 To present relevant adds to each consumer as determined by a recommendation system.

3.4 Social Networks

The promise of social networks is to match-make a group of consumers with a shared set of interests or profiles. A recommendation system can determine group of consumers who are alike in what they want to share. In this scenario, the products refer to other people whom a consumer would like to interact with. A recommendation system can seamlessly achieve this grouping with the now-classical message: **"People like you..."**.