A Unified Approach for Context-sensitive Recommendations

Mihaela Dinsoreanu, Florin Cristian Macicasan, Octavian Lucian Hasna and Rodica Potolea
Computer Science Department, Technical University of Cluj-Napoca, Gh. Baritiu st. 26-28, Cluj-Napoca, Romania

Keywords: (Dynamic) Content Context Match, Classification, Topic Model, Parallelization, Text Mining, Taxonomy, Design and Implementation, Evaluation.

Abstract: We propose a model capable of providing context-sensitive content based on the similarity between an analysed context and the recommended content. It relies on the underlying thematic structure of the context by means of lexical and semantic analysis. For the context, we analyse both the static characteristics and dynamic evolution. The model has a high degree of generality by considering the whole range of possible recommendations (content) which best fits the current context. Based on the model, we have implemented a system dedicated to contextual advertisements for which the content is the ad while the context is represented by a web page visited by a given user. The dynamic component refers to the changes of the user's interest over time. From all the composite criteria the system could accept for assessing the quality of the result, we have considered relevance and diversity. The design of the model and its ensemble underlines our original view on the problem. From the conceptual point of view, the unified thematic model and its category based organization are original concepts together with the implementation.

1 PROBLEM STATEMENT

Nowadays, in an information centric society, we are flooded with data (the so called "deluge of data" (The Economist, 2010)). In this context, an automatic identification of entities that satisfy the user's information needs is paramount. Our goal is to expose only meaningful information (relevant and of interest) to the user. More important it has to be at the right time and in the right context (Garcia-Molina et al., 2011).

We propose a model capable of providing context-sensitive content based on the underlying thematic similarity between an analysed context and the recommended content. We expose this relation with the help of a unified topic model that extracts the topics describing both the context and the content. This process is fuelled by the portions of the analysed entities having the highest descriptive value. Once these entities are annotated with thematic information, their reciprocal affinity, within the unified topic model, can be measured. Using such values, a topic based coverage aims to improve diversity and achieve serendipitous recommendations.

Our main objectives are:

- Extracting the highest descriptive valued n-grams among the analysed entities;
- Attaching thematic information to the analysed entities;
- Maximizing diversity of the recommended content.

We will adopt, as proof of concept, the online advertising's problem of the best match (Broder and Josifovki, 2011) between an active context (web page), suitable content (advertisements) and the user that is currently interacting with that context, which will be referred as OABM. We argue that this problem can be mapped on our model by using a double instantiation of the context-to-content similarity relation. One instantiation describes the relation between a web page representing the context and an advertisement that maps on the content. The second has the same mapping for the content but describes the context as being the user, moreover his/her historical information. The combined, triple recommendation between an active context, the content and the dynamic context is constructed by further processing the two instantiations.

2 STATE OF THE ART

A common approach in literature is to describe the

Dinsoreanu M., Macicasan F., Hasna O. and Potolea R..
A Unified Approach for Context-sensitive Recommendations.
DOI: 10.5220/0004141700850094
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2012), pages 85-94
ISBN: 978-989-8565-29-7

matching content with relevant keywords. These keywords are compared with the descriptors of the ads (bid phrase) hence obtaining a lexical similarity (Manning et al., 2008); (Yih et al., 2006). Such an approach follows a pipeline with a few, well-defined stages. A pre-processing stage is proposed (Yih et al., 2006) to prepare the content by sanitizing, removing stop words, stemming and extracting some keyword candidates (words from the context, annotated with some descriptive features). Then the annotated keyword candidates are processed, in a Monolithic Combined approach, by a binary classifier. This is how the keywords are selected and the keyword selection step is completed.

Such an approach is generally enhanced with additional models that sustain the semantic similarity between web pages and advertisements (Broder, 2007); (Zhang et al., 2008); (Ribeiro-Neto et al., 2005). This association generates a semantic score which, combined with the lexical, consolidates the match. This semantic information can be embedded in a taxonomy (Broder, 2007) and used to score the similarity based on the distance to the least common ancestor, if both the context and the advertisement can be mapped on it.

The third aspect to be considered in such a model consists of the particularities of an actual user. The associated historical information, if present, will influence the final match (Ahmed et al., 2011) (Chakrabarti et al., 2008). User information can be attached to the advertisement or to the page (Chakrabarti et al., 2008) but, recent research explores the idea of user interest and behavioural trend (Ahmed et al., 2011). Such a model can extract the dynamics of behaviour and make better recommendations.

The concept of a "topic" is described using a specialized *mixed membership model* called *topic model.* Such a model describes the hidden thematic structure (Blei, 2011) in large collections of documents. The Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is such a topic model. Its distinguishing characteristic is that all documents in the collection share the same set of topics, but each document exhibits those topics with different proportions (Blei, 2011). These topics are defined by a distribution over the whole set of available words (within the document corpus). The documents are described by a distribution over topics. This distribution was subconsciously modelled by the author of the document, who intended to transmit a message (collection of words) about an area of interest (the document's distribution over topics) using area specific words (distribution of topics over words). All the used words are just sampled from the topic's word distribution. In a model like that, the only observable data are the document's words. The topics, their distribution in documents and the distribution of words between topics need to be inferred. Direct inference is not tractable so approximation techniques are used (Heinrich, 2009).

Many systems produce highly accurate recommendations, with reasonable coverage, yet with limited benefit for practical purposes (such as buying milk and bread from the supermarket), new dimensions that consider the "non-obviousness" should be considered. Such dimensions are coverage (percentage of items part of the problem domain for which predictions are made), novelty and serendipity, dimensions for which also (Ziegler et al., 2005) advocates. Since serendipity is a measure of the degree to which the recommendations are presenting items that are attractive and surprising at the same time, it is rather difficult to quantify. "A good serendipity metric would look at the way the recommendations are broadening the user's interests over time" (Herlocker et al., 2004), so, again the need for introducing timing and sequence of items analysis for RS.

3 TERMINOLOGY

In the following we formally define the notions used throughout the article.

- A *word* is the basic unit of discrete data (Blei, et al., 2003), defined to be an element of a vocabulary V;
- A *topic* $\beta_t = \{ p(w|t) \mid w \in V\}$ is a probability distribution over a finite vocabulary of words where $\sum_{i \in \beta_t} i = 1$. Let T be the set of all topics;
- A *document* $d = \{w_1, w_2, \dots, w_N\}$ is a sequence of N words. Each document has an associated distribution over topics $\theta_d = \{p(t|d) | t \in T\}$ where $\sum_{i \in \theta_d} i = 1$;
- A *keyword* k_d is an element of a document having high descriptive value. Let $K_d = keywords(d)$ represent all the keywords of a document;
- A *context* C_x is a specialized document which is not necessary to have a well-defined structure;
- A *dynamic context* ΔC_x is a specialized context that evolves over time due to system interactions;
- An *active (*or *analysed) context* $C_x A$ is a specialized context analysed during an iteration;
- A content C_n is a specialized document that is to

be associated with a context;

- A *corpus* $C = \{d_1, d_2, \ldots, d_D\}$ is a collection of D documents;
- A *unified topic model* is a 5-tuple $UTM = \langle V, T, \{\beta_t | t \in T\}, C, \{\theta_d | d \in C\}\rangle$ describes the set of all the topics T, their distribution over words β_t, the underlying vocabulary V, all the documents C and their distribution over topics θ_d;
- A *category* ψ is an *abstraction* over the *topic* adding semantic value. It has an associated distribution over topics denoted θ_ψ and is organized in a *category taxonomy* Ψ;
- An *contextual relevance* $rel(C_xA, C_n)$ measures the similarity of the analyzed context with a content;
- A *dynamic relevance* $rel(\Delta C_x, C_n)$ measures the similarity of the dynamic context with content.

4 SYSTEM ARHITECTURE

Functional Description of the Modules: Our architecture (Figure 1) consists of four main modules that interact to generate recommendations.

The first module, *Keyword Extractor* (KE), identifies and extracts the elements of the context with the highest descriptive value. Those elements represent *keywords,* which outline the significance within the analysed context. This module performs a pre-processing step that prepares the candidates for *keyword* status by annotating them with the features used in the classification step. The result of this module is a set of n-grams that best describe (summarize) the initial context. They are used as an input by the next module. Formally, KE is described as follows:

$$\mathrm{KE}(C_X A) = K_{C_X A}. \tag{1}$$

The *Topic Identifier* (TI) is responsible for associating topic information to the analysed context $(C_X A)$ based on the keywords that describe it$\left(K_{C_X A}\right)$. The association is accomplished using the TI's underlying *topic model*. At this point the topic level unification takes place by associating to $C_X A$ a distribution over topics $\theta_{C_X A}$. From now on, all the analysed entities are modelled by a distribution over topics within the *unified topic model*. Formally, TI is described as follows:

$$\mathrm{TI}\left(K_{C_X A}\right) = \theta_{C_X A}. \tag{2}$$

The *Category Combiner* (CC) is responsible for computing the similarity between the topic distribution generated by TI $(\theta_{C_X A})$ and the distribution associated to the managed content (θ_{C_n}) or dynamic context $(\theta_{\Delta C_x})$. The main limitation of the topics discovered by TI is anonymity (topics have no semantic information). To overcome this shortcoming we added an abstraction layer above the topics called category. Such categories are nodes in a taxonomy having a pre-computed topic distribution. In CC we also analyse the dynamic context that describes the evolution of the interaction based on previously acquired data. The output of this module is a set of advertisements (Υ) with two associated relevance values. One is computed from the perspective of the active context ($rel(C_xA, C_n)$) and the other, from the perspective of the dynamic context ($rel(\Delta C_x, C_n)$). Formally, CC is described as follows:

$$\mathrm{CC}\left(\theta_{C_x A}, \theta_{\Delta C_x}\right) = \Upsilon, \tag{3}$$

where

$$\Upsilon = \left\{ \left(\begin{matrix} C_n, \\ \langle rel(C_x A, C_n), \rangle \\ rel(\Delta C_x, C_n) \end{matrix} \right) \middle| C_n \in \psi \right\}. \tag{4}$$

With the constant growth of online data, Recommendation Systems face the problem of dealing with huge information spaces. Thus selecting a small representative subset of items, which are not just simply relevant for the user, but at the same time offer an element of novelty, is rather difficult. The diversity measure of the system is focusing to offer users the pleasant surprise of finding an unexpected item of great interest, beyond "obviousness". Overall it is searching for a solution preserving the high quality of retrieved items obtained and offering diversity, therefore serendipity of results.

The *Ranker* (R) is responsible for filtering the output according to the actual performance criteria (Γ). Such criteria can range from relevance to diversity or trustworthiness. In the case of diversity, we aim a low thematic overlap between recommendations while maintaining their relevance to the considered context (whether it is C_xA or ΔC_x). This reduced overlap induces an increased context thematic coverage that is more likely to produce serendipitous recommendations. Formally, R is described as follows:

$$\mathrm{R}(\Upsilon) = \left\{ C_n \middle| C_n \in \underset{C_n \in \Upsilon}{\operatorname{argmax}}\, \Gamma \right\}. \tag{5}$$

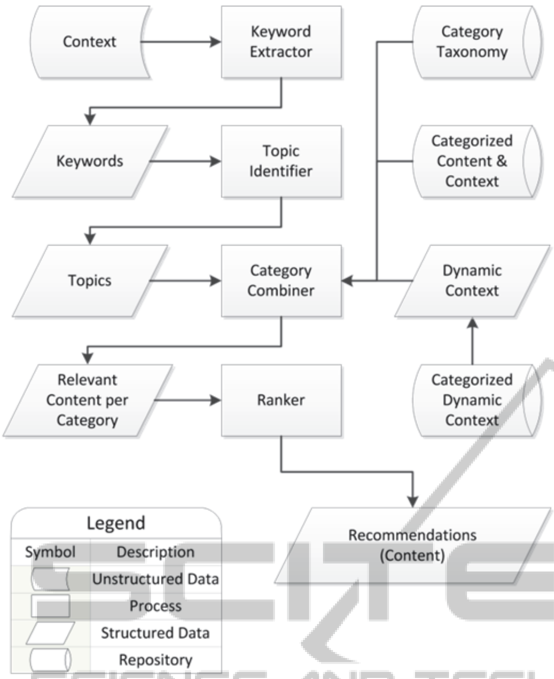

Figure 1: Conceptual architecture.

5 CONTEXTUAL ADVERTISEMENT CASE STUDY

We instantiate the model to propose a system capable of providing a context-sensitive recommendation to a user based on the triple similarity relation between the active context, the advertisement and the user's historical information.

The OABM problem can be decomposed in two *context-to-content* relations defined based on the same underlying *unified topic model*. Due to this common reference, the values describing the similarity within the context-to-content relation can be extrapolated, at model level. This way, the two relations unify in a high level, OABM independent topic model. We formally specialize the generic terms to OABM specific concepts:

- A *web page* C_xW is a specialized active context;
- An *advertisement* C_nA is a specialized content;
- A *user interaction* is a specialized dynamic context that is described by its *overall interest* I.
- The *contextual relevance* maps at OABM level to $rel(C_xW\ ,C_nA)$;
- A *behavioural relevance* $rel(I\ ,C_nA)$ is a specific dynamic relevance that describes the user's similarity with an advertisement.

The active context is represented by a web page that can be described by a set of keywords extracted from it. The advertisements have associated bid phrases that are considered keywords. We consider, as historical information, the context (both web page and advertisement) with which the user interacts. This interaction defines the user's current interest. This is why the keywords that describe the context can also describe the current user interest. For a personalized content-to-context match, the dynamic characteristics of the interests are considered.

From the conceptual point of view, we employ a unified technique for the recommendation of relevant advertisements. We extend the concept of a *topic* to describe all the three components of OABM. An advertisement is described by a *single*, targeted *topic*. The active context is modelled by a *static* set of *topics*. A user is described by a *dynamic* set of *topics* due to the evolution of interests over time. Thus, the tree components can be defined, at topic level, based on a common reference i.e. *the unified topic model* in Figure 2.

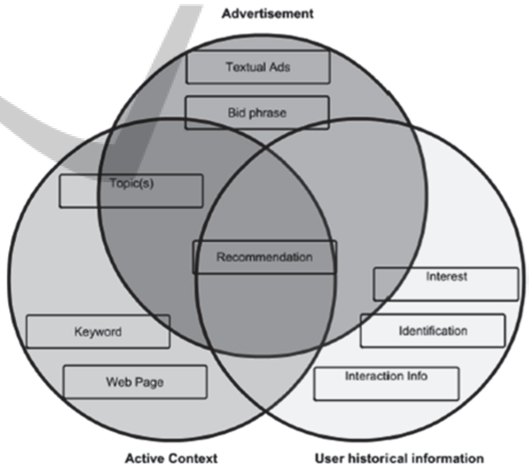

Figure 2: The unified topic model.

We define $LDA_e(\{\vartheta|\vartheta \in keywords(e)\}): \theta_c$ as a function accepting as input the set of keywords describing the entity *e* and returning its distribution over topics θ_e.

We define the active's context topic distribution as:

$$\theta_{C_xA} = LDA_{C_xA}(\{\vartheta|\vartheta \in keywords(C_xA)\}). \quad (6)$$

We also define the *content* topic distribution as:

$$\theta_{C_n} = LDA_{C_n}(\{\vartheta|\vartheta \in keywords(C_n)\}). \quad (7)$$

We claim that $\exists\ t \in T$ s.t. $\text{argmax}_t(\text{p}(t|C_n)) = \{t\}$ where $\text{p}(t|C_n) \in \theta_{C_n}$ hence the content is described by a dominant topic.

The dynamic set of interests is defined based on three lever hierarchy: short (I_s), medium (I_m) and long (I_l) term (Ahmed et al., 2011).

We define the user's overall interest (I) based on a convex combination between the three sub-interests. Let $\kappa \in (0,1)$ and $\mathrm{K} = \kappa + \kappa^2 + \kappa^3$ such that:

$$I = \frac{\kappa}{\mathrm{K}} * I_s + \frac{\kappa^2}{\mathrm{K}} * I_m + \frac{\kappa^3}{\mathrm{K}} * I_l \qquad (8)$$

Now let $I_i,\ i \in \{s, m, l\}$ be one of the three sub-interests and $C_i = \{\varepsilon | visit(\varepsilon) \in interval(i)\}$ the set of all accessed contexts during the interval associated with the sub-interest. At this level, we may employ the following definition for $I_i, i \in \{s, m, l\}$:

$$I_i = LDA_{C_i}\left(\bigcup_{c \in C_i} keywords(c)\right) \qquad (9)$$

and describes the sub-interest as the distribution over topics of a pseudo-context described by the union of the keywords associated to a context accessed during the sub-interest interval.

5.1 Keyword Extractor

In this step we perform the processing needed to extract the components of the context with the highest descriptive value. Its internal structure is presented in Figure 3.

The **Feature Extraction** (*FE*) sub-module is responsible for the initial pre-processing of the analysed context. A succession of steps is performed to bring the context's elements in an annotated candidate state. We first perform *stop-word removal* and *stemming*. The remaining candidates are enhanced with features that underline their status within the context. We associate to each candidate its occurrences statistic information together with other characteristics dependent on the context's nature like the candidate's localization within the context, its styling information or its inner structure. The *candidates with features* generated by the FE sub-module are persisted in a repository.

The **Keyword Selection** (*KS*) sub-module uses a binary classifier for selecting the keywords from the candidates. The classifier is chosen based on the specific criteria required by CA. From the business point of view an advertisement recommendation that is out of context is worst then no advertisement at all. Thus, the specific need of our problem is to increase precision as much as possible, by allowing moderate degradation of recall.

The keyword classification process needs to use features that differentiate between the components of the analysed text and filter out bad candidates. Such features can range from statistical descriptors of the occurrences of a word based on both its local and global statistics to localization markers or unique style definitions. Their ultimate goal is to best describe the classification category to which the membership relation is in question.

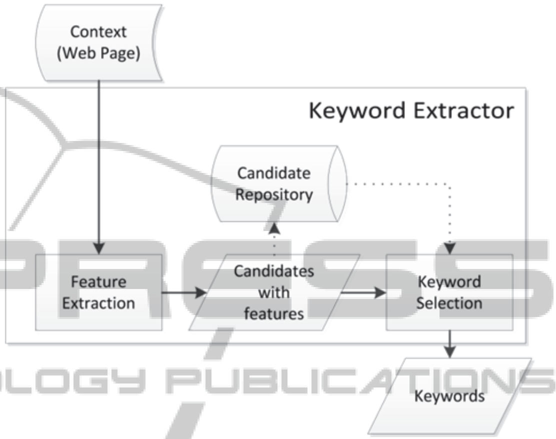

Figure 3: Keyword extractor detailed view.

5.2 Topic Identifier

The *Topic Identifier* (TI) module employs a method of extracting the thematic structure in a corpus of documents.

The Latent Dirichlet Allocation generative model proposed in (Blei et al., 2003) is formally described by the graphical model in Figure 4. This model describes a corpus of D documents on which a number of K topics are defined with a β_k word distribution for topic k.

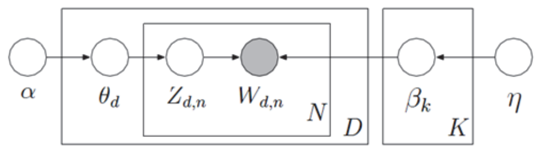

Figure 4: Graphical model representation of LDA.

Each document has a θ_d distribution over topics from which, for each of the N words in the document, a topic $Z_{d,n}$ is sampled followed by the sampling of the word $W_{d,n}$ from $\beta_{Zd,n}$. The α parameter controls the sparsity (Heinrich, 2009) of θ while η influences β. A decrease of α causes the model to characterize documents with a few topics but high weight. A decrease of η means that the model prefers to assign fewer words to each topic.

Shadowed nodes define an observable variable hence all the information we have are the words $W_{d,n}$. The rest need to be inferred.

Direct inference of the latent (unobservable) variables is not tractable (Heinrich, 2009). This is why approximation techniques must be used. We chose to employ a *collapsed Gibbs sampling* approximation technique (Griffiths and Steyvers, 2002).

We adopted, as a starting point, the Java implementation proposed in (Phan and Nguyen, 2008) and extended it to a *parallel Gibbs sampling LDA* implementation. We employed an input decomposition technique by dividing the documents analysed during each of the Gibbs iteration in evenly distributed work packages for the parallel processes. We use a shared memory model. Its consistency is maintained by a synchronization mechanism that controls the access to the critical section in which the underlying topic model is updated. Using such a technique we obtain an average 1.85 relative speed-up for 2 processing elements.

5.3 Category Combiner

The CC module is responsible for computing the similarity between the analysed context and both the managed advertisements and user's interests based on their distribution over topics within the *unified topic model.* The internal structure of CC is presented in Figure 5.

The **Category Assigner** (CA) receives as input the distribution of topics for the given context and a taxonomy of categories (with their topic distribution) to construct the mapping between a category and a topic. In order to quantify the similarity between two probability distributions we use the Hellinger distance (Nikulin, 2011). We search categories that minimize the distance to the input distribution. The module returns the list of candidate categories ordered by their relevance. A subset of these categories is processed by the next module.

These categories are further combined with the *dynamic context* to select a final subset of advertisements that qualify for the next processing step.

The *dynamic context* generation is a process enforced by the **Dynamic Behaviour Modeller** (DBM). This sub-module aggregates the user information (in the form of interests). This information is used to enable behavioural recommendations.

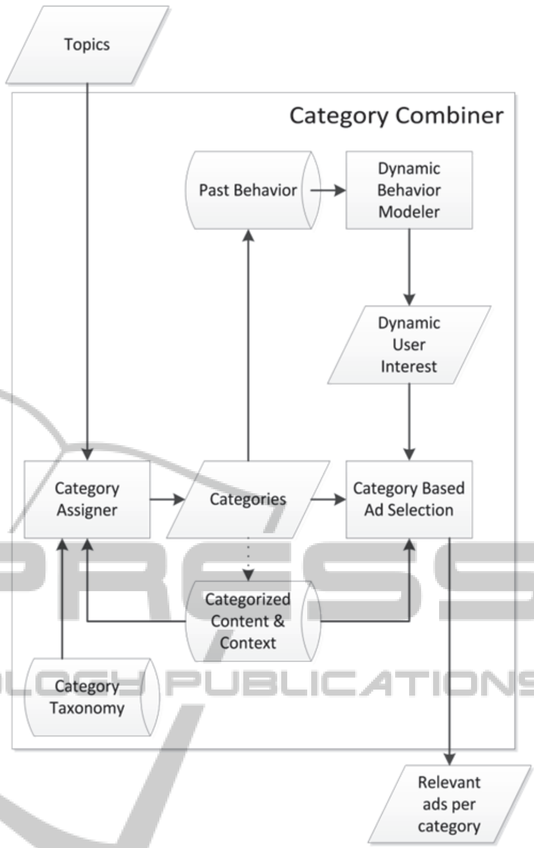

Figure 5: Category combiner detailed view.

The **Category-Based Ad Selector** analyses ads associated to categories, in a reduced search space due to the category set cardinality reduction yielded by CA and DBM. We further compute, for each advertisement, two similarity scores based on the Hellinger distance between their probability distributions. The first score is related to the contextual relevance and the second to the behavioural relevance. Both scores will be further integrated in the ranking module.

5.4 Ranker

At the level of R, we have the possibility of recommending advertisements from two different perspectives due to the double instantiation of the context-to-content relation. The first, that maximizes the similarity between the *topics* describing the active context and the advertisement, is considered to be a *contextual* recommendation and will influence the page topic coverage. The second, that considers the best match between the user's interests and an advertisement, named a *behavioural* recommendation (Ahmed et al., 2011) and it improves the user interest coverage. The *page topic coverage* and *user interest coverage* may be antagonistic by nature because they compete for the same page advertisement slots. Balancing them by choosing the ones with the maximal relevance in the given overall context is recommended.

These two approaches are specific to the marginal values of the *correlation coefficient* λ. We represent the overall recommendation as:

$$\begin{aligned}\text{similarity}_{\lambda}(C_xW, I, C_nA) \\ = rel(C_xW\ , C_nA) * \lambda \\ +\ rel(I\ , C_nA) *\ (1-\lambda).\end{aligned} \tag{10}$$

In (10) the triple similarity relation $\text{similarity}_{\lambda}(C_xW, I, C_nA)$ is represented as a convex combination of the behavioural relation and the contextual relation. Moreover, the behavioural recommendation maps on $\text{similarity}_0(C_xW, I, C_nA)$ and the contextual recommendation maps on $\text{similarity}_1(C_xW, I, C_nA)$.

6 RESULTS AND EVALUATION

We chose the classifier candidates from the ones offered by Weka (Witten et al., 2011). For the experiment we used a 1333 instance subset of the dataset with 75%/25% class distribution. The training and test subsets were selected randomly with a 66% Train/Test Percentage Split with a total of 10 repetitions.

Table 1: Classifier comparison for the Keyword Selection step.

Name	Correct Classification	Precision	False Positive Rate	F-measure
Bayes Network	89.73	0.96	0.12	0.93
Naïve Bayes	86.44	0.92	0.24	0.91
Multilayer Perceptron	91.17	0.95	0.14	0.94
SMO	86.24	0.90	0.31	0.91
Bagging (J48)	92.04	0.96	0.13	0.94
Decision Table	89.40	0.93	0.22	0.93
J48	91.70	0.95	0.14	0.94
Decision Stump	79.20	1.0	0.0	0.84
AdaBoost M1	87.81	0.93	0.22	0.92
SPegasos	87.70	0.92	0.24	0.92

The results in Table 1 show that the best performing classifiers are the Multilayer Perceptron (MLP), J48 and a Bagged Predictor with underlying J48 (BJ48). The MLP's training time is greater than the others with at least an order of magnitude and grows with the dataset. Hence it is discarded as a candidate. The remaining candidates have the same underlying classifier but BJ48 performs additional replications and voting with a minimal improvement of the target measurements. Thus our final choice is the J48 classifier.

The largest computational effort appears during the computation of the topic distributions. This process is dependent on the number of words that describe the topic model hence it is in a perpetual growth as we acquire new data. We chose to adopt a parallel implementation for this critical area of the flow. We measured the relative speedup (ΔS) while varying the dimension of input parameters like the number of topics to be discovered (#T), the number of iterations to approach convergence (#I) and the number of analysed documents (#D) for the estimation and inference (inf.) use-cases. For the experimental results covered in Table 2 we used two processing elements. Further investigation showed proportional growth of ΔS as the number of processing elements increases.

Table 2: Improvement of parallel LDA.

Use case	#T	#I	#D	Sequential [s]	Parallel [s]	ΔS
Estimation	30	20	2246	5.52	3.05	1.81
	50	20	2246	8.02	4.40	1.82
	50	40	2246	16.26	8.74	1.86
	100	40	2246	30.39	16.46	1.83
	100	100	2246	77.72	40.47	1.92
	50	40	1123	8.23	4.40	1.87
Inf.	100	100	1123	25.38	14.03	1.81

We can observe that the growth of a single measure of interest with a controlled increment will generate a proportional growth in both sequential and parallel results by maintaining the relative speedup in a constant range. But when we increase multiple measures of interest with significant increments we observe a spike in the relative speedup hence favouring the parallel implementation.

Two processing elements prove to bring a significant boost for our needs, but we would like to further evaluate our approach while considering an increasing number of such resources. We further considered a fixed workload scenario where we varied the processing units. Our findings are summarized in Figure 6. We observe 82% efficiency by the time we consider four processing elements and 70% as we get to eight, our available maximum. In a highly parallelized environment, intensive topic model interactions will generate contention on our critical section that makes us slowly converge to our Amdahl limit.

Another aspect of interest is represented by the benefit introduced by the usage of categories as an additional abstraction layer above topics.

Figure 7 shows the evolution of execution time as the number of analysed contexts grows both with and without the usage of categories. This behaviour appears in the CA sub-module of the CC. Our category taxonomy has 100 nodes organized on 6 levels. We can see that even if the category based approach starts with a default overhead, as the number of advertisements grows, the two curves will intersect when the total number of advertisements equals the number of categories combined with the number of ads in the top categories and from that point on, their growth patterns differ with almost a *decade*.

Another important aspect is the evaluation of the provided recommendations. Due to the subjective nature of the underlying problem and to the lack of annotated benchmark datasets we chose to employ an end user evaluation of the recommended advertisements. A comparison with similar systems from literature is unavailable because researchers do not publish their datasets. We considered the manual inspection performed by a group of users that were asked to assess, on 1-10 rating scale, the similarity of a content with a designated context. The representative results are presented in Table 3.

Table 3: User evaluation compared with thematic similarity.

Context	Content (ad)	User Average Points (UAP)	Hellinger Distance (HD)	HD per UAP
C27	A26	8	0.63	0.079
	A910	9	0.57	0.064
	A867	2	0.88	0.441
C42	A283	3	0.93	0.309
	A736	6	0.77	0.130
	A882	7.5	0.70	0.099
C54	A801	2	0.86	0.427
	A884	6	0.77	0.127
	A128	2	0.88	0.438

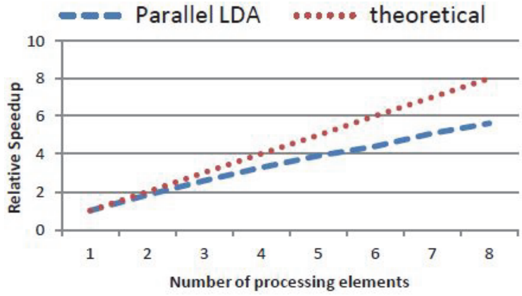

Figure 6: Relative speedup evolution.

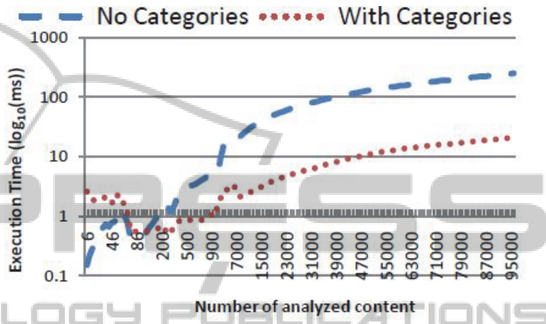

Figure 7: Evolution of category-based selection.

We aim to minimize Hellinger distance, hence lower values are welcomed. This is why the lower the HD per UAP is the better. We can see good recommendations like A26 and A910 for C27 having an 8 and 9 UAP with smaller than 0.1 HD per UAP. This shows a good correlation between the user evaluation and the results of the Hellinger distance.

Moreover we explore the influence of the *correlation coefficient* λ on the recommendations. We assess the marginal cases for which we have fully contextual or fully behavioral recommendations and the case in which the two are combined. We consider a user and a web page having their top three topics illustrated in Table 4.

Table 4: Top three topics of analysed context (user and web page).

	Topic1		Topic2		Topic3		Combined Coverage
User (U1)	T21	25%	T22	17%	T44	50%	92%
Web Page (C27)	T5	35%	T27	18%	T48	15%	68%

The topics in Table 4 cover, in different proportions, the context but their combined coverage is sometimes enough to describe them. We consider a set of recommended advertisements for which we compute the degree of coverage and the combined similarity, using different versions of the correlation coefficient (λ).

Table 5: Full behavioural (λ=0) recommendation comparison.

Content	T21 (%)	T22 (%)	T44 (%)	Coverage (%)	HD to User	Rank
A26	4	12	21	13.05	0.88	3
A910	8.8	80	8.8	22.05	0.68	1
A867	32	14	9.4	15.40	0.73	2

Table 6: Full contextual (λ=1) recommendation comparison.

Content	T5 (%)	T27 (%)	T48 (%)	Coverage (%)	HD to Web Page	Rank
A26	11.2	16.7	35	12.17	0.63	2
A910	26.4	44.4	8.8	18.5	0.57	1
A867	4.7	40	4.7	9.52	0.88	3

Table 7: Combined (λ=0.5) recommendation comparison.

Content	HD2User	HD 2 WP	HD2Overall	Rank
A26	0.88	0.63	0.75	2
A910	0.78	0.57	0.71	1
A867	0.86	0.88	0.83	3

We observe that, there is a correlation between the low values of the Hellinger distance and the associated coverage from both the perspectives.

This correlation spiked our interest for exploring what would be the relation between the Hellinger distance and the user's evaluation. To this purpose we constructed the curve representing the relation between the score of an association and its distance. Figure 8 depicts our findings. At the lowest end of the evaluation interval (ranging between 0 and 4) one can observe a strong variation between the actual distance and the score. But at this level is difficult to assess the correlation because users see content within this band from different perspectives. The important aspect is that content in this range is not desirable. On the other hand, at the other extremity of the interval one can observe an evolution of the distance to ever-decreasing values. The area of confusion is between 4 and 7. Here, the distance varies with small increments making it harder to discriminate.

We concluded that Hellinger values above 0.9 have a higher chance of being *bad* content and values below 0.7 of being *good* content. If we consider that this measure is theoretically bound between 0 and $\sqrt{2}$ this results seems promising.

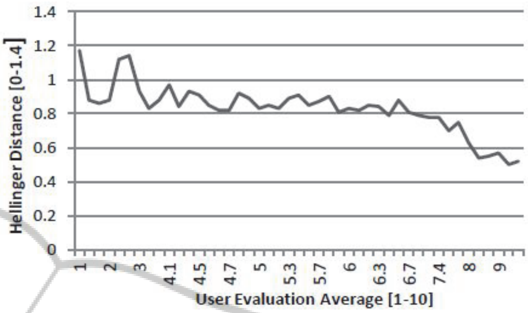

Figure 8: Hellinger evolution with user evaluation.

7 CONCLUSIONS

This paper proposes a model capable of providing context-sensitive content based on the similarity relation between the analysed context and the recommended content.

The similarity is measured at thematic level by mapping both the context and the content within a common reference system, the proposed unified topic model. In order to achieve topic information, we analyse the underlying thematic structure induced by the keywords of the considered entities using a parallel topic extraction approach. We also consider the influence of continuous system interactions by modelling them as the dynamic context. The content search space is reduced using the category abstraction and the quality is measured based on relevance and diversity.

We applied the model on the OABM problem considering as context a web page, as content an advertisement and as dynamic context the user interacting with them.

We are currently evaluating feature selection mechanisms in order to identify a (near) optimal feature set. We are also considering other approximation techniques to tackle the intractability of the inference associated with LDA.

We aim to integrate serendipity as a metric in our ranking module. Thus we need means of quantifying this desirable feature. In this regard we intend to explore the use of the NDGG-IA compound metric (Agrawal and Gollapudi, 2009) as a measure of serendipity in new, original approaches for subset selection. Moreover, following the ideas presented in (Santos et al., 2010), we intend to define and evaluate the relative relevance of an item to the set of the user's needs (optimal set) and the needs covered by the retrieved items set.

REFERENCES

Agrawal, R. & Gollapudi, S., 2009. Diversifying Search Results. Barcelona, *WSDM*.

Ahmed, A. et al., 2011. Scalable distributed inference of dynamic user interests for behavioral targeting. San Diego, California, USA, *ACM*, pp. 114-122.

Blei, D. M., 2011. Introduction to probabilistic topic models. *Communications of the ACM*, December, 54(12), pp. 77-78.

Blei, D. M., Ng, A. Y. & Jordan, M. I., 2003. Latent dirichlet allocation. T*he Journal of Machine Learning Research*, 1 March, Volume 3, pp. 993-1022.

Broder, A. a. F. M. a. J. V. a. R. L., 2007. A semantic approach to contextual advertising. Amsterdam, The Netherlands, *ACM*, pp. 559-566.

Broder, A. & Josifovki, V., 2011. *Introduction to computational advertising (MS&E 239)*. Stanford(California): Stanford University.

Chakrabarti, D., Agarwal, D. & Josifovski, V., 2008. Contextual advertising by combining relevance with click feedback. Beijing, China, *ACM*.

Garcia-Molina, H., Koutrika, G. & Parameswaran, A., 2011. Information seek-ing: convergence of search, recommendations and advertising. *Communications of the ACM*, 54(11), pp. 121-130.

Griffiths, T. L. & Steyvers, M., 2002. *A probabilistic approach to semantic representation*. Fairfax, Virginia, s.n., pp. 381-386.

Heinrich, G., 2009. *Parameter estimation for text analysis*, Darmstadt, Germany: Fraunhofer IGD.

Heinrich, G., 2009. P*arameter estimation for text analysis*, Darmstadt, Germany: Fraunhofer IGD.

Herlocker, J. L., Konstan, J. A., Terveen, L. G. & Riedl, J. T., 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, January, 1(22), pp. 5-53.

Manning, C. D., Raghavan, P. & Schtze, H., 2008. I*ntroduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.

Nikulin, M., 2011. *Hellinger distance - Encyclopedia of Mathematics*. [Online] Available at: http://www.encyclopediaofmath.org/index.php?title=Hellinger_distance&oldid=16453 [Accessed 22 April 2012].

Nikulin, M., 2011. H*ellinger distance - Encyclopedia of Mathematics*. [Online] Available at: http://www.encyclopediaofmath.org/index.php?title=Hellinger_distance&oldid=16453 [Accessed 22 April 2012].

Phan, X.-H. & Nguyen, C.-T., 2008. *JGibbLDA: A Java implementation of latent Dirichlet allocation (LDA)*. [Online] Available at: http://jgibblda.sourceforge.net/ [Accessed 22 April 2012].

Ribeiro-Neto, B., Cristo, M., Golgher, P. B. & Silva de Moura, E., 2005. Impedance coupling in content-targeted advertising. Salvador, Brazil, *ACM*, pp. 496-503.

Santos, R. L., Macdonald, C. & Ounis, I., 2010. Exploiting query reformulations for web search result diversification. Raleigh, North Carolina, USA, *ACM*, pp. 881-890.

The Economist, 2010. *The data deluge*. [Online] Available at: http://www.economist.com/node/15579717 [Accessed 22 April 2012].

Witten, I. H., Frank, E. & Hall, M. A., 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd ed. San Francisco(CA): Morgan Kaufmann Publishers Inc..

Yih, W.-t., Goodman, J. & Carvalho, V. R., 2006. Finding advertising keywords on web pages. Edinburgh, Scotland, *ACM*, pp. 213-222.

Zhang, Y., Surendran, A. C., Platt, J. C. & Narasimhan, M., 2008. Learning from multi-topic web documents for contextual advertisement. Las Vegas, Nevada, USA, *ACM,* pp. 1051-1059.

Ziegler, C.-N., McNee, S. M., Konstan, J. A. & Lausen, G., 2005. Improving recommendation lists through topic diversification. Chiba, Japan, *ACM*, pp. 22-32.