Abstract—We consider the problem of matching the topics of a scientific paper with those of possible publication venues for that paper. While every researcher knows the few top-level venues for his specific fields of interest, a venue recommendation system may be a significant aid when starting to explore a new research field. We propose a venue recommendation system which requires only title and abstract, differently from previous works which require full-text and reference list: hence, our system can be used even in the early stages of the authoring process and greatly simplifies the building and maintenance of the knowledge base necessary for generating meaningful recommendations. We assessed our proposal using a standard metric on a dataset of more than 58000 papers: the results show that our method provides recommendations whose quality is aligned with previous works, while requiring much less information from both the paper and the knowledge base.

Index Terms— Recommending systems; Latent Dirichlet Allocation; n-grams

I. INTRODUCTION

Publishing a research paper is the main goal of every researcher. Choosing the right venue where to submit a paper depends on several factors: venue reputation, venue topics, whether to submit to a journal or a conference, location and date of conferences.

Assessing the reputation of a scientific venue automatically is a long-standing problem, for which many solutions have been proposed and is still a subject of a vigorous debate [1]. In this work, we focus on the problem of matching the topics of a paper with those of publication venues. This is a key factor for increasing the likelihood of receiving sound reviews and may help in bringing a research work to the attention of researchers working on similar topics, thereby improving its potential in terms of future citations.

While every researcher knows the few top-level venues for his specific fields of interest, there are several practical scenarios in which choosing the right venue is difficult, for example when starting to explore a new research field. For example, in Computer Science alone there are more than 2000 venues [2]. Many of them are highly specific, but many others are quite generalist and yet many others occupy different positions along the broad spectrum between those two extremes. It is virtually impossible for any researcher to have both high precision and recall about all those venues and their corresponding topics. A system capable of recommending possible publication venues for a paper could thus be a real aid to many researchers. Indeed, a few proposals of this sort have started to emerge in the recent years [2], [3], [4].

In this work, we propose a topic matching procedure that can form the basis of a recommendation system for scientific paper submission. The best performing existing proposals require the full-text of the paper to be examined, including the list of references and of authors, while our approach requires only title and abstract. This peculiarity of our proposal is important because it allows querying the system even in the early stages of the authoring process and because it may greatly simplify the building and maintenance of the knowledge base necessary for generating meaningful recommendations.

We developed and assessed three variants based on techniques that are proven to be highly effective in text classification: Latent Dirichlet Allocation and n-gram based Cavnar-Trenkle classification. We performed an experimental evaluation using the standard metrics for recommendation systems, on a dataset of more than 58000 papers extracted from the Microsoft Academic Search engine. The results show that our method provides recommendations whose quality is aligned with the existing state of the art, while requiring much less information from both the paper and the knowledge base.

II. RELATED WORK

Recommender systems are used to automatically suggest one or more items to the user from a set of items. They became more and more useful as the amount of information available to the users grew. Recommender systems are successfully used to suggest movies, news, tags, and so on, basing on different techniques [5].

In the recent years, much work has been done in the field of recommender systems for research papers: [6] shows that over 80 different approaches (presented in more than 170 research papers, patents and web pages) have been proposed in the last 14 years. Yet, only a tiny fraction of them (3 on 80) concern the specific task of venue recommendation [2], [3], [4].

Our proposal differs from all of the cited works in terms of the kind and amount of information required in order to provide a recommendation for a paper: we only require the paper abstract and title and do not need supplementary information such as full-text, references, citation or authorship. Hence, our
system may be used in an earlier stage of the research life-
cycle, when that supplementary information is not available.
Moreover, recommender systems which require also citation
data need databases including citations, which has been shown
to have a significantly lower coverage compared to text-only
(authors, title and abstract) databases [7], [6]; eventually, those
system accuracy is negatively affected.

In [2], a system is proposed which is based on Collaborative
Filtering—a technique which is widely used in recommending
systems. A set of features is computed for each paper which
contains both content and stylometric features. Similarly to
our proposal, in the cited work content features consist of
paper distribution over 100 topics, obtained using the Latent
Dirichlet Allocation (LDA) [8]. Stylometric features are a
set of 300 context-free features including lexical (number of
words, average sentence length, and alike), syntactic (number of
function words, count of punctuation, and alike) and structural
(number of sections, figures, and alike) features: it follows that
most of these features are meaningful only when extracted from
the full-text. These features are then used to compare distances
from the paper to be examined and choose the n closest papers—
n going from 500 to all papers. The venue which occurs most
frequently among the closest papers is finally recommended.
The authors also propose a method improvement which weights
the closest papers venues according to the relation with the
paper to be examined (i.e., cited by, authored by at least
one common author, and alike). The experimental evaluation—
performed on two large datasets totaling about 200000 papers—
shows that both the use of stylometric features and relation-
weights do indeed increase accuracy.

In [4], a method is proposed for accomplishing differ-
ent recommendation tasks for research papers, including
recommendation of other similar papers, suitable reviewers
and publication venues. The proposed method is actually
implemented in a publicly available web application\(^1\) whose
architecture is described in [9]. The goal of the proposed
method is to augment researchers ability in performing a
literature search. To this end, a researcher provides the system
with a set of papers (seed) and receives back an enlarged sets
including other related papers. The system can also be used as
a publication venue recommender if the seed is the set of
papers cited in the paper to be examined: indeed, this is the
way the authors evaluate their proposal in that specific task.
The proposed system bases on the citation graph and
does not take into account paper text: the rationale is that
text may include ambiguities—i.e., same concepts denoted by
different terms—and hence make the recommendation less
effective. The cited paper presents different techniques: the
best performing one is a modified version of Random Walk
with Restart technique (RWR) which also considers the graph
direction (DARWR, Direction-aware RWR). This modification
is useful to tune a search in order to promote either more
recent or traditional relevant papers. Yet, the authors do not
show if and how the modification is exploited in the task of
publication venue recommendation.

In [3], a method is shown which bases on the author
network analysis. Given a paper for which only the author
names are required, a social graph of is built (by crawling the
Microsoft Academic Search website) where a node corresponds
to an author and an edge is drawn between two nodes if the
corresponding authors co-authored at least one paper, up to
the third level. Then the venue which occurs more frequently
among the papers appearing in the graph is recommended. An
obvious limitation is that each paper authored by the same set
of authors will receive the same recommendations, regardless
of the actual paper topic. Three variants of the method are
proposed: in the best performing one, the venues occurring
in the graph are weighted according to the weight of edges,
i.e., the number of times two authors co-authored a paper.
The authors evaluate their proposal on a very small dataset,
including only 16 venues and less than 1000 papers.

III. OUR APPROACH

A. Scenario

Let \( V = \{ v_1, v_2, \ldots \} \) be a predefined set of publication
venues. The problem consists in generating, given a new paper
\( a \), a recommendation list \( (v_1, \ldots, v_N) \) of suitable publication
venues for \( a \), \( N \) being a configurable parameter, where the
list is ordered from the most suitable to the least suitable.
We describe in Section IV-B the metric which we use for
quantifying this notion.

We propose three different recommendation methods in the
following sections. Each method requires a preliminary learning
phase to be performed only once based on a knowledge base
of papers already published in the venues in \( V \). In the actual
recommendation phase, the recommendation lists for papers
not available in the learning phase are generated.

In each method the representation of a paper \( a \) consists of the
concatenation of the paper title, abstract and keywords, which
is then pre-processed as follows: (i) convert to lowercase;
(ii) replace all digits with a single space; (iii) replace all
punctuation with a single space; (iv) remove leading, trailing
and multiple spaces; (v) remove all words whose length is
lower than 3 characters; (vi) remove common English stop
words; (vii) perform a stemming.

B. Cavnar-Trenkle

This method is based on a long-established text classification
method [10], which has been shown to be able to correctly
discriminate between different languages and different subjects.

In the learning phase, a n-gram profile is built for each
venue \( v \in V \), as follows. Let \( A_v \) be a set of papers published
at the venue \( v \). For each paper \( a \in A_v \), we extract and count
its n-grams up to length 5, i.e., all the subsequences of \( a \)
which do not include spaces or line termination characters
and whose length is between 1 and 5 characters, included.
Then, for each resulting n-gram, we sum its counts over all the
\( a \in A_v \). Finally, we sort the n-grams according to their counts,
in decreasing order, and truncate the resulting list to \( n_{ng} = 300 \)
items. We set the n-gram profile \( p_v \) of venue \( v \) to the truncated

\(^1\)http://theadvisor.osu.edu
The underscore character _ represents the space character: it occurs often in the profile because of the pre-processing described in the previous section, which replaces punctuation and digits with a character.

In the recommending phase, the n-gram profile \( p_\text{ng} \) of the paper \( a \) to be examined is computed as above. Then, for each venue \( v \in V \), we compute a profile distance \( d \) between \( p_\text{ng} \) and \( p_v \) as follows. Initially \( d = 0 \); for each \( n \)-gram \( x \in p_\text{ng} \), we increment \( d \) by \( |i_v - i_a| \), where \( i_v \) and \( i_a \) are the positions of \( x \) in \( p_v \) and \( p_\text{ng} \), respectively; in case \( x \notin p_v \), we increment \( d \) by \( n_\text{ng} \). For example, the profile distance between \( p_v = \{ a, b, c, c, c, c \} \) and \( p_\text{ng} = \{ d, d, c, c, c, a \} \), with \( n_\text{ng} = 3 \), is 6. Finally, we recommend the \( N \) venues with the lowest profile distances from \( p_\text{ng} \).

### C. Two-steps-LDA

This method is based on the concept of probabilistic topic model and, in particular, on Latent Dirichlet Allocation (LDA) [8]. LDA is a generative probabilistic model for a collection of texts. The model assumes the existence of a predefined set of topics and a predefined set of words. Topic probabilities are defined over the collection of texts and word probabilities are defined over each topic. A given text in the collection is considered to have been generated by first drawing a distribution of the topics and then a distribution of the words for each topic.

In [8], a method is also proposed to compute the posterior of the generative probabilistic model, given a collection of texts. In this method LDA may be seen as a black-box which works in two operating modes.

In collection mode, LDA receives as input a set \( \{ a_1, a_2, \ldots \} \) of papers and a value for a parameter \( k \)—the predefined number of topics. In this work, we set the number of topics to 20, as this value seems to be a reasonable estimate for the number of main topics in Computer Science. We remark that only the number of topics is to be defined in advance: topics need not be specified as “names” or list of words. In collection mode LDA outputs: (i) for each topic, its word probabilities, i.e., a vector \( w_j = (w_{j,1}, w_{j,2}, \ldots) \) with one element for each word found in the set of papers; \( w_{j,i} \) is the probability of the \( i \)-th word to appear in a paper related to the \( j \)-th topic; (ii) for each paper \( a_j \in A \), its topic probabilities, i.e., a vector \( t_j = (t_{j,1}, \ldots, t_{j,k}) \) with one element for each topic; \( t_{j,i} \) is the probability that the \( j \)-th paper is related to the \( i \)-th topic.

In item mode, LDA receives a single paper \( a \) and the vectors of word probabilities associated with each of the \( k \) topics: \( w_1, \ldots, w_k \). LDA outputs the vector \( t \) which represents the topic probabilities for \( a \).

We use this method as follows. In the learning phase, we apply LDA in collection mode to all the papers in \( A = \bigcup_{v \in V} A_v \) and assign a single prevalent topic to each \( v \in V \). In detail, (i) we assign a single topic to each paper \( a \in A_v \), i.e., the topic with highest probability in the vector \( t \) associated with \( a \); (ii) we count the topic assignments for all the papers \( a \in A_v \), and assign to \( v \) the topic with highest count. For example, Table II shows the topic assignments for a conference of our dataset including 200 papers: for ease of understanding, we include in the table the 4 most probable words for each topic (the words with the highest \( w_{j,i} \)—those words depend only on the topic, not on the specific conference. By assigning a main topic to each conference, we partition the venues in \( V \) according to their prevalent topic. We denote by \( V_i \subset V \) the set of all venues whose assigned topic is \( i \) (it might be \( V_i = \emptyset \) for one or more topics \( i \)). We set the number of main topics \( k_{\text{nd}} = 20 \).

Then, we assign a prevalent subtopic to each venue. To this end, we apply again LDA in collection mode, separately for papers in each partition \( V_i \) of venues (i.e., we apply again LDA once for each topic, each time only with papers in venues for which that topic is the prevalent one). We set

2There are different figures about the number of topics in Computer Science research, which is estimated to be 14 in [11], 27 in [12] and 17 in [13]. Microsoft Academic Search divides the Computer Science domain in 24 non mutually exclusive sub-domains: i.e., there are venues which appear in more than one sub-domain.
the number of subtopics $k_d = 20$. We associate with each
venue $v$ also a subtopic probabilities vector $t_v$. This vector is
the average of the topic probabilities of the papers in $A_v$, i.e.,
$$t_v = \frac{1}{|A_v|} \sum_{a \in A_v} t_a.$$ During the learning phase we also saved
all the corresponding word probabilities $(k_m(1 + k_d)$ vectors).

In the recommending phase, we apply LDA in item mode
to the paper $a$ to be examined (using the word probabilities
of the main topics found above) and obtain its corresponding
vector of topic probabilities $t_m$. We assign to $a$ the topic $i_m$
with highest probability in $t_m$. If $V_{i_m} = \emptyset$, we recommend no
venues for $a$. Otherwise, we apply LDA in item mode to $a$
(using the word probabilities of the subtopics of the topic $i$),
obtain $t_i$ and assign a subtopic $i_s$ to $a$. Then, we select the
subset $V_{i_m,i_s}$ of $V_{i_m}$ which contains all the venues whose main
topic is $i_m$ and subtopic is $i_s$. Finally, we recommend the first
$N$ venues of $V_{i_m,i_s}$ whose average subtopic vector $t_v$ is the
closest (by means of Euclidean distance) to $t_i$. Note that, when
using this method, we could recommend less than $N$ venues
for a paper.

D. LDA+clustering

This method is based on LDA as the previous one, but also
clusters papers according to their topic probabilities.

In the learning phase, we apply LDA in collection mode
to all the papers of $A$ with $k_m = 20$ and obtain, for each paper
$a$, a vector $t_a$; in other words, we associate a point in $[0, 1]^{km}$
with each paper. We then cluster the papers point in $k_c = 12$
clusters using the k-means clustering method—we chose this
value after preliminary experimentation and evaluation of the
Silhouette index [14] for $8 \leq k_c \leq 50$. We hence partition the
set of all papers according to their cluster index: we denote with $A_i$ the set of papers of the $i$-th cluster.

Then, for each cluster $i$, we apply LDA in collection mode
to the papers of $A_i$ with $k_m = 20$. Let $V_i$ be the set of venues
for which at least one paper belongs to $A_i$: we associate
with each $v \in V_i$ an average subtopic vector $t_v$ which is the
average of the topic probabilities of the $v$ papers in $A_i$, i.e.,
$$t_v = \frac{1}{|V_i|} \sum_{a \in V_i} t_a.$$ In the recommending phase, we apply LDA in item mode
to the paper $a$ to be examined (using the word probabilities obtained
from LDA application to all $A$ papers) and obtain $t_m$. Then,
we choose the cluster $i$ whose centroid is the closest (by means of Euclidean distance) to $t_m$. We apply again LDA in item mode to $a$ (using the word probabilities obtained from
LDA application to $A_i$ papers) and obtain $t_i$. Finally, we recommend the first $N$ venues of $V_i$ whose average subtopic vector $t_v$ is the closest (by means of Euclidean distance) to $t_i$. Note that, as for the previous method, we could recommend
less than $N$ venues for a paper.

E. Method motivations

The rationale for the three methods are as follows.

With the Cavnar-Trenkle method, we assume that each venue exhibits a specific language profile, shaped by the papers previously published at that venue. Then, we recommend the venues whose language profiles are the closest to the language profile of the examined paper.

With the Two-steps-LDA method, we assume that each venue is associated with exactly one main topic and one subtopic. Then, we recommend the venues whose main topic and subtopic match with the main topic and subtopic of the paper to be examined.

Finally, with the LDA+clustering, we assume that all the papers may be clustered according to the mix of main topics they are about—we could consider each cluster as a research field; moreover, each venue may publish papers which possibly belong to different fields. Then, we recommend the venues whose average subtopics mix are the most similar to the subtopic mix of the paper to be examined, provided that some of the papers they previously published belong to the same field of the paper to be examined.

IV. EXPERIMENTAL EVALUATION

A. Dataset

We composed a dataset of about 58000 papers, using the
Microsoft Academic Search engine (MAS), as follows. We
selected the Computer Science domain and queried the engine
for the 300 conferences which published at least one paper in
the last 5 years (2008 to 2012 included), sorted by decreasing
Field Rating—Field Rating is a metric defined by MAS which is
similar to h-index and assesses the impact of a venue or
author within its specific field. Then, for each conference, we
queried MAS for the last 200 published papers (including those published before 2008) and discarded those for which
the abstract field was empty. At the end, we collected a dataset
$A$ of 58466 papers partitioned almost uniformly among 300
conferences.

MAS defines 24 sub-domains for the Computer Science do-
main and associates each venue with at most three sub-domains

\[\text{http://academic.research.microsoft.com}\]
We performed a 2-fold evaluation procedure, as follows. We partitioned \( A \) into \( A_1 \) and \( A_2 \), such that both partitions contained the same number of papers for each of the 300 conferences. Then, for each recommendation method, we performed the learning phase on \( A_1 \) followed by the recommendation phase for each paper \( a \in A_2 \); we repeated the procedure after swapping \( A_1 \) and \( A_2 \).

Table IV shows three recommendations obtained with our system for three papers of the dataset described above. The first (topmost) paper received as first recommendation the venue at which it was actually published, but also the other two venues appear to be suitable. The actual venue was not recommended for the second and third papers; yet, it can be seen that in both cases the first recommended venue appears to be suitable.

We assess recommendations with the standard metric used in earlier works [2], [3], [4], i.e., venue-accuracy@\( N \) defined as the ratio between the number of correct recommendations and the number of all recommendations. Let \( v_a \) denote the ground-truth venue at which paper \( a \) was actually published. A recommendation for paper \( a \) is correct if and only if \( v_a \) is among the \( N \) venues recommended by the method under evaluation.

We also computed the sub-domain-accuracy@\( N \) used in [3]. According to this metric a recommendation for paper \( a \) is correct if and only if at least one of the \( N \) recommended venues is associated with one of the sub-domains associated with \( v_a \).

Sub-domain-accuracy@\( N \) is a weaker metric than venue-accuracy@\( N \), as it requires the ability to match 1 sub-domain on 24. On the other hand, venue-accuracy@\( N \) could be excessively and unnecessarily severe, as it assumes that the papers composing our dataset have been published to the most suitable venue, in terms of research topic matching. This assumption does often not hold, as there are many factors which affect how authors choose venues, such as conference date, location, reputation and so on.

We compare our results with those obtained by previous works [2], [3], [4]. However, since those works have been evaluated using datasets which differ in terms of number of papers and venues (and this affects the corresponding accuracies), we also provide a simple baseline which corresponds to the accuracy obtained with a random recommender, i.e., a recommender which suggests \( N \) venues chosen at random. Concerning the venue-accuracy@\( N \), the random recommender simply exhibits an accuracy of \( \frac{1}{N} \) / 300. Concerning the sub-domain-accuracy@\( N \), the random recommender accuracy computation can be estimated as \( 1 - (1 - p)^N \), where \( p = \frac{1}{20} \) is the probability of matching the ground-truth sub-domain with exactly one venue guess—\( p \) takes into account that, in our dataset, most venues (about 80%) are related to exactly one sub-domain, while the others are related to two or three sub-domains.

**C. Results and discussion**

Table V shows the results of our experimentation in terms of venue- and sub-domain-accuracy@\( N \) averaged on the two folds, for \( N \in \{3, 5, 10\} \). The table also shows the corresponding figures for the random recommender and the three previous works for the same venue recommendation task, where available.

It can be seen that both Cavnar-Trenkle and LDA+clustering methods can provide recommendations which appear significantly better than those of the random recommender. Their venue-accuracy@\( N \) is an order of magnitude greater than the baseline for all values of \( N \); it is 45.6% and 33.2% for Cavnar-Trenkle and LDA+clustering respectively.

The Two-step-LDA performs only slightly better than the baseline; this result concurs with the finding of [2], where a trivial LDA-only method is used as baseline and provides very low accuracy (1.8% on venue-accuracy@10 on ACM data, against 79.8% obtained with the best method proposed in the cited work). We agree with those authors and think that recommendations based only on topic models build on textual content may suffer terminology ambiguities: on the other hand, we show that different techniques which do not involve LDA or augment LDA outcome exhibit a significantly greater accuracy, while do not relying on other than abstract and title.

Concerning the comparison against the other previous works, the Cavnar-Trenkle method is only slightly less accurate than [2] (considering the average of the two datasets used in the cited work): 45.6% vs. 49.4% for \( N = 10 \) and 34.0% vs. 39.8% for \( N = 5 \). The performance gap with respect to [4] is larger. In assessing these results it is important to remark that our approach requires only title and abstract, while [2], [4] require citation information and/or full-text (see Section II). It is fair to
Table IV

Some publication venue recommendation obtained with our system (Cavnar-Trenkle method). The second column shows the first three recommendations and, in italic, the actual venue of the paper.

<table>
<thead>
<tr>
<th>Title and abstract fragment</th>
<th>Recommendations (N = 3)</th>
</tr>
</thead>
</table>
| **High-frequency shape and albedo from shading using natural image statistics** | 1. Computer Vision and Pattern Recognition  
2. Storage and Retrieval for Image and Video Databases  
3. International Conference on Computer Vision |
| **An efficient community detection method using parallel clique-finding ants** | 1. International Conference on Weblogs and Social Media  
2. Recent Advances in Intrusion Detection  
3. IEEE INFOCOM (IEEE Congress on Evolutionary Computation) |

Table V

The recommendation accuracy obtained with our methods, the random recommender and 3 previous works—for these, a dash (-) is shown where an experimental evaluation is not available. Last two columns show the size of the dataset for the experimentation as reported in the cited works: n.a. means that the figure is not provided.

<table>
<thead>
<tr>
<th>Method</th>
<th>Venue-acc. @ N (%)</th>
<th>Sub-domain-acc. @ N (%)</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N = 3</td>
<td>N = 5</td>
<td>N = 10</td>
</tr>
<tr>
<td>Cavnar-Trenkle</td>
<td>26.8</td>
<td>34.0</td>
<td>45.6</td>
</tr>
<tr>
<td>Two-step-LDA</td>
<td>3.4</td>
<td>3.8</td>
<td>4.0</td>
</tr>
<tr>
<td>LDA+clustering</td>
<td>16.1</td>
<td>21.7</td>
<td>33.2</td>
</tr>
<tr>
<td>Random recommender</td>
<td>1.0</td>
<td>1.7</td>
<td>3.3</td>
</tr>
<tr>
<td>[2] ACM</td>
<td>-</td>
<td>55.7</td>
<td>69.8</td>
</tr>
<tr>
<td>[2] CiteSeer</td>
<td>-</td>
<td>23.9</td>
<td>29.0</td>
</tr>
<tr>
<td>[3]</td>
<td>91.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>[4]</td>
<td>-</td>
<td>-</td>
<td>63.2</td>
</tr>
</tbody>
</table>

Note, though, that [2], [4] experiment with a dataset containing a larger number of venues, which likely makes the resulting scenario more challenging. In this respect, the proposal [3] only requires authorship information but is exercised on a very small dataset: 960 papers from 16 conferences across 3 years. That proposal is assessed using sub-domain-accuracy, but with only 4 sub-domains (corresponding to 4 ACM Special Interest Groups). A random recommender would obtain a sub-domain-accuracy @3 of 1 - (1 - \frac{1}{4})^3 = 57.8\%, which suggests that the considered scenario is poorly challenging.

The above results have been obtained with a single-threaded prototype implementation written in R and run on commodity hardware (notebook with quad-core 3GHz cpu and 4GB ram). The learning phase took 4 min, 50 min and 25 min respectively for the Cavnar-Trenkle, Two-step-LDA and LDA+clustering methods (applied to 29233 papers); the recommending phase took 0.5 s, 1.6 s and 1.7 s for one paper.

V. Concluding Remarks

We have proposed a topic matching procedure that can form the basis of a recommendation system for scientific paper submission. Key feature of our proposal is that it requires only title and abstract of the paper. This feature may be very important in practice, from the point of view of both users (the system may be queried even in the early stages of the authoring process) and developers (building and maintaining the knowledge base is much simpler than required by earlier proposals).

We have assessed our proposal experimentally on a large and challenging dataset composed of 58000 papers from 300 conferences. We have demonstrated that title and abstract may suffice for generating recommendations which are indeed meaningful and whose quality is aligned with the existing state of the art. Our analysis suggests that recommendations built upon long-established n-gram based text classification methods may be highly effective, while recommendations based on generative and probabilistic topic models may lead to unsatisfactory results. The proposed system is feasible also from a performance point of view, as the learning phase requires a few minutes while a recommendation may be generated in a couple of seconds.

Of course, our proposal needs further investigation and, in this respect, our results should be validated in other domains beyond Computer Science.

References


