Significant Words Representations of Entities

SIGIR-DC 2016 Submission (Long Version)

Under supervision of Dr. Jaap Kamps

Mostafa Dehghani
Institute for Logic, Language and Computation
University of Amsterdam, The Netherlands
dehghani@uva.nl

ABSTRACT

Transforming the data into a suitable representation is the first key step of data analysis, which the performance of any data oriented method is heavily dependent on it. This need is concerned with questions surrounding how we can best learn representations for textual entities that are: 1) precise, 2) robust against noisy terms, 3) transferable over time, and 4) interpretable by human inspection. In this research, we propose *significant words language models* of a set of documents that capture all, and only, the significant shared terms from these documents. This is achieved by adjusting the weights of terms already well explained by the document collection as well as the weight of terms that are only explained by specific documents, which eventually results in having the significant terms left in the model.

Our main contributions are the following. First, we define significant words language models as an iterative estimation process, resulting in effective models capturing the essential terms and their probabilities. Second, we apply the resulting models to several applications like group profiling, feedback problem, and hierarchical classification and see a better performance over the state-of-the-art methods. Third, we see that the estimation method is remarkably robust making the models insensitive to the noisy terms and transferable during the time and at the same time interpretable by human inspection. The proposed approach is generally applicable to other systems that require the estimation of an effective model representing significant features of a group of objects.

CCS Concepts

•**Information systems → Document representation; Language models;** *Personalization; Query reformulation; Clustering and classification;*

Keywords

Significant Words Language Model; Parsimonious Language Model

1. INTRODUCTION

Transformation of raw data to a representation that can be effectively exploited is motivated by the fact that data oriented methods often require input that is convenient to process. Regarding the fact that, real-world data is usually complex, noisy, and highly variable, it is necessary to discover data representations that are less affected by non-essential features [1].

In this research we introduce *significant words language models* (SWLM) as a family of models aiming to learn representations for the set of documents so that all, and only, the significant shared terms are captured in the models. This makes these models to be not only distinctive, but also supported by all the documents in the set. We refer to a set of documents as an “entity” which could be an indicator of a person, an organization, a category, an ideology, and so on.

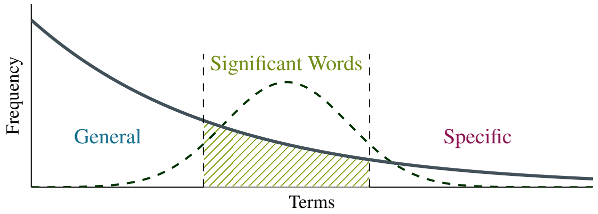

Figure 1: Establishing a set of “Significant Words” based on Luhn [14]

The general idea of our research is inspired by the early work of Luhn [14], in which he argues that to extract **significant words**, we need to avoid both common observations and rare observations. More precisely, Luhn assumed that frequency data can be used to measure the significance of words to represent documents. Considering Zipf’s Law, he simply devised a counting technique for finding significant words. He specified two cut-offs, an upper and a lower (see Figure 1), to exclude insignificant words.

Pursuing the Luhn’s idea, parsimonious language models has been proposed by Hiemstra et al. [10] to make the standard language models more distinctive by eliminating the effect of common terms from the model employing a more advanced way than using fixed frequency cut-offs. However, unlike Luhn, they do not take the risk of excluding rare words since some of these words are strongly discriminative and removing them makes the models less distinguishable.

Our research in a way completes the cycle, following the vision of Luhn. We introduce a meaningful translation of specificity and generality against significance and propose an effective way of establishing a representation consisting of significant words, by *parsimonizing* the model toward not only the common observations, but also the rare observations.

Generally speaking, SWLM tries to estimate language models from the set of documents which is “specific” enough to distinguish the features of these documents from other documents by removing general terms, and at the same time, “general” enough to capture all the shared features of these documents, by excluding document specific terms. To do so, SWLM assumes that terms in the each document in the set are drawn from three models: 1. *General model*, representative of common observation, 2. *Specific model*, representative of partial observation, and 3. *Significant Words model* which is a latent model representing the significant characteristics of the whole set. Then, it tries to extract the latent significant words model.

In the rest of this article, we outline our research questions in Section 2. Then, in Section 3, we discuss details of our proposed model. Section 4 describes the applications of our model and briefly reports on evaluation results. Finally, we discuss our future direction as well as our challenges and open questions in Section 5.

2. RESEARCH QUESTIONS

The main goal of this research is *to define and estimate significant representations for multiple entities that are not affected by neither general properties nor specific properties.* We break down this into number of concrete research questions:

RQ1 *How to estimate SWLM from a set of documents capturing all, and only, the essential shared commonalities of these documents?*

RQ2 *What are the different applications of SWLM and how effective is it in these applications?*

RQ3 *How can SWLM be used as an analytical tool, which gives key insights into the characteristics of the data?*

RQ4 *How to apply the idea of SWLM in other environments for example on the output of embedding methods to improve the final representations?*

In the following sections, we address some of these questions and we discuss about our plan to investigate some of them in the future of this research.

3. ESTIMATING SWLM

In this section, we address our first research question, **RQ1**: "How to estimate SWLM from a set of documents capturing all, and only, the essential shared commonalities of these documents?"

In order to estimate SWLM for a set of documents, we assume that there are three models from which each document in the set is generated as a mixture sampling from these models: *significant words* model, *general* model, and *specific* model. The significant words model represents the latent model as the distribution of terms reflecting the essential characteristics of the set. The general and specific models, however, are not necessarily topic-centric models. In a way, they are supposed to represent distribution of terms that are not considered as significant information. In order to extract these two models, patterns of the occurrences of terms in different documents are taken into consideration. In loose terms, the general model represents very common observed terms and the specific model represents the partially observed terms, which we assume as two different patterns of distribution of insignificant terms.

Each model is represented using a terms distribution, or a unigram language model, θ_{sw}, θ_g, and θ_s. Based on the generative model, each term in a document is generated by sampling from a mixture of these three models independently. Thus, the probability of appearance of the term t in the document d is as follows:

$$p(t|d) = \lambda_{d,sw} p(t|\theta_{sw}) + \lambda_{d,g} p(t|\theta_g) + \lambda_{d,s} p(t|\theta_s), \qquad (1)$$

where $\lambda_{d,x}$ stands for $p(\theta_x|d)$ which is the probability of choosing the model θ_x given the document d.

Based on the patterns of term occurrences in the documents as external knowledge, we estimate θ_g and θ_s and make them fixed in the estimation process as infinitely strong priors. We consider the collection model, θ_C as an estimation for θ_g:

$$p(t|\theta_g) = p(t|\theta_C) = \frac{c(t,C)}{\sum_{t' \in V} c(t',C)}, \qquad (2)$$

where $c(t,C)$ is the frequency of term t in the collection. This way, terms that are well explained in the collection model get high probability and are considered as general terms.

Furthermore, we establish a definition for "specificity" with regards to our main goal which is estimating a representation for a set of documents, as being supported by part of the documents in the set but not all. We estimate θ_s to represent the probability of a term being partially observed as follows, and normalize all the probabilities to form a distribution:

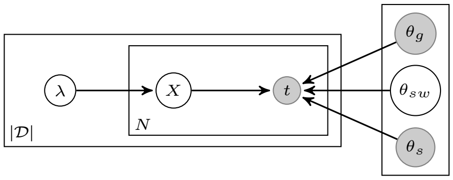

Figure 2: Plate diagram of SWLM.

$$p(t|\theta_s) = \sum_{d_i \in \mathcal{D}} \Bigg(p(t|\theta_{d_i}) \prod_{\substack{d_j \in \mathcal{D} \\ j \neq i}} (1 - p(t|\theta_{d_j})) \Bigg), \qquad (3)$$

where $P(t|\theta_{d_i}) = {}^{c(t,d_i)}/_{\sum_{t' \in d_i} c(t',d_i)}$. Intuitively, Equation 3 considers the probability of term t to be a specific as it's being important in one of the document models but not others, marginalizing over all the documents in the set. This way, terms that are well explained in only one document but not others get higher probabilities and are considered as insignificant specific terms.

Having the above assumptions, the goal is to fit the log-likelihood model of generating all terms in the documents to discover the term distribution of the significant words model, θ_{sw}. Let $\mathcal{D} = \{d_1, \dots, d_{\mathcal{D}}\}$ be the set of documents. The log-likelihood function for the entire set of feedback documents is:

$$\log p(\mathcal{D}|\Upsilon) = \sum_{d \in \mathcal{D}} \sum_{t \in V} c(t,d) \log \Big(\sum_{x \in \{sw,g,s\}} \lambda_{d,x} p(t|\theta_x) \Big), \qquad (4)$$

where $c(t,d)$ is the frequency of the term t in the document d, and Υ determines the set of all parameters that should be estimated, $\Upsilon = \{\lambda_{d,sw}, \lambda_{d,g}, \lambda_{d,s}\}_{d \in \mathcal{D}} \cup \{\theta_{sw}\}$.

To fit our model, we estimate the parameters using the maximum likelihood (ML) estimator. Therefore, assuming that documents are represented by a multinomial distribution over the terms, we solve the following problem:

$$\Upsilon^* = \operatorname*{argmax}_{\Upsilon} p(\mathcal{D}|\Upsilon) \qquad (5)$$

Assuming that $X_{d,t} \in \{sw, g, s\}$ is a hidden variable indicating which model has been used to generate the term t in the document d, we can compute the parameters using the Expectation-Maximization (EM) algorithm. The stages of the EM algorithm are as follows:

E-Step

$$p(X_{d,t} = x) = \frac{p(\theta_x|d) p(t|\theta_x)}{\sum_{x' \in \{sw,g,s\}} p(\theta_{x'}|d) p(t|\theta_{x'})} \qquad (6)$$

M-Step

$$p(t|\theta_{sw}) = \frac{\sum_{d \in \mathcal{D}} c(t,d) p(X_{d,t} = r)}{\sum_{t' \in V} \sum_{d \in \mathcal{D}} c(t',d) p(X_{d,t'} = r)} \qquad (7)$$

$$\lambda_{d,x} = p(\theta_x|d) = \frac{\sum_{t \in V} c(t,d) p(X_{d,t} = x)}{\sum_{x' \in \{sw,g,s\}} \sum_{t \in V} c(t,d) p(X_{d,t} = x')} \qquad (8)$$

Figure 2 represents the plate notation of SWLM. As it is shown, for each document the contribution of each of three models, λs, are estimated. It can be seen that general model, θ_g, and specific model, θ_s are considered as external observations, which are involved in the estimation process as infinitely strong priors.

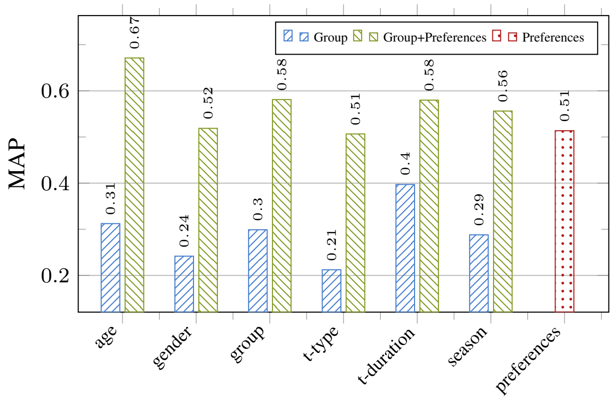

Figure 3: Performance of employing user preferences-based and group-based customization on contextual suggestion task.

4. MODEL'S APPLICATIONS

In order to assess the effectiveness of SWLM, we have employed it in different applications. In the following sections, we briefly discuss these applications, prompting **RQ2**: "What are the different applications of SWLM and how effective is it in these applications?"

4.1 Group Profiling

In this section, we address the question: "*How to employ SWLM on group profiling and how effective are the models on content customization?*"

Group profiling is to understand and model the characteristics of the group of objects. One of the important applications of group profiling is in the content customization, which generally is the process of tailoring content to individual users' characteristics or preferences. In the content customization, using individual preferences is not always possible. For example, sometimes there is a new user in the system with no historical interactions and no rich information about the preferences, or sometimes the user is not able to determine his/her preferences explicitly. In these situations, group based content customization would be beneficial to suggest content to the user based on the preferences of the groups that the user belongs to.

We propose to use SWLM to extract the 'abstract' group level latent model that captures all, and only, the essential features of the whole group. We have employed the resulting models in the task of contextual suggestion. Analysing different grouping criteria using TREC 2015 contextual suggestion[1] batch task dataset, we find that group-based suggestions using SWLM improve the performance of content customization [4, 8]. Figure 3 reports the results of one of our experiments on evaluating the performance of group-based suggestion employing different grouping approaches, individual preferences-based suggestion, and combinations of these two approaches.

4.2 Feedback

One of the applications in which applying SWLM leads to built a better model is the Feedback problem. In this section we address the question"*How to employ SWLM on (pseudo) relevance feedback? How does it prevent the feedback model to be affected by non-relevant terms of non-relevant or partially relevant feedback documents?*"

The main goal of feedback systems is to extract a feedback model from a set of feedback documents, where the model represents the "relevant" documents. However, the existence of documents with a broader topic or multiple topics in the feedback set (both in relevance feedback and pseudo-relevance feedback) can distract the feedback model by adding bad expansion terms, leading to *topic drift* [9]. Using SWLM for estimating feedback model enables us to tackle this challenge and extract a language model of feedback documents capturing the essential terms representing the *mutual notion of relevance*, i.e the representation of relevance that is not only distinctive, but also supported by all the feedback documents.

Table 1: Performance of different systems on pseudo relevance feedback on different datasets. Baseline methods are the maximum likelihood estimation —without feedback (MLE) [12], the simple mixture model (SMM) [18], the relevance models (RM3) [13], and maximum-entropy divergence minimization model (MEDMM) [15]. ▲ indicates that the improvements over all other runs are significant at the 0.05 level using the two-tailed t-test.

Method	**Robust04**		**WT10G**		**GOV2**	
	MAP	*P@10*	*MAP*	*P@10*	*MAP*	*P@10*
MLE	0.2501	0.4253	0.2058	0.3031	0.3037	0.5147
SMM	0.2745	0.4381	0.2087	0.3159	0.3140	0.5163
RM3	0.2732	0.4626	0.2291	0.3215	0.3245	0.5236
MEDMM	0.2842	**0.4700**	0.2308	0.3258	0.3311	0.5287
RSWLM	**0.2874**	0.4681	**0.2407**▲	**0.3346**	**0.3421**▲	**0.5366**

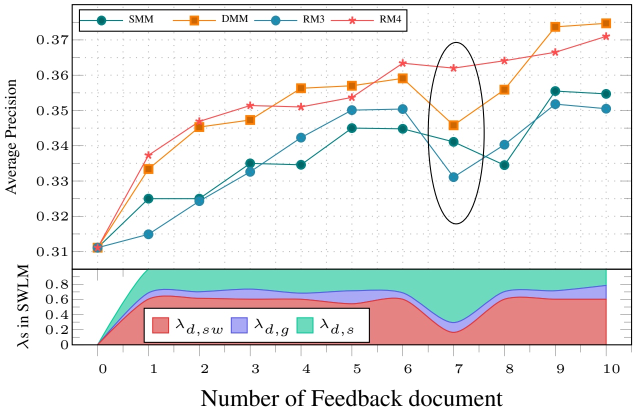

Figure 4: Dealing with poison pills: Effectiveness of different feedback systems facing with bad relevant document in topic 374 of TREC Robust04.

It is obvious how SWLM can be used for estimating language model from the set of feedback documents. However, in the original estimation process, information from the query is considered for estimating the feedback model. In order to involve information from the original query, inspired by the work by Tao and Zhai [16], we modify the estimation process and incorporate the extra knowledge from the query model by defining a prior parameter and employ maximum a posteriori to fit the model to feedback documents and solve the following problem:

$$\Upsilon^* = \operatorname*{argmax}_{\Upsilon} p(\mathcal{D}|\Upsilon)P(\Upsilon) \tag{9}$$

We define the a conjugate Dirichlet prior on θ_{sw} as follows:

$$p(\theta_{sw}) \propto \prod_{t \in V} p(t|\theta_{sw})^{\beta p(t|\theta_q)}, \tag{10}$$

where $\beta p(t|\theta_q)$ is the parameter of the Dirichlet distribution which in fact performs as the additional pseudo-count for t to push the model θ_{sw} to assign a higher probability to term t as it has a high probability in θ_q. We call the new model *Regularized SWLM* (RSWLM). Generally speaking, RSWLM adds a bias in the estimation process to bend the feedback model toward the query model.

[1] https://sites.google.com/site/treccontext/trec-2015

```
Estimating SWLM
procedure EstimateSWLMs
    Initialization:
    for all objects $o$ in the hierarchy do
        $\theta_o \leftarrow$ Standard estimation for $o$
    repeat
        Specification
        Generalization
    until Models do not change significantly anymore
```

(a) Overall procedure of estimating SWLM.

```
Specification Stage
procedure Specification
    Queue ← all objects in breadth first order
    while Queue is not empty do
        $o \leftarrow$ Queue.pop()
        $l \leftarrow o$.Depth()
        while $l > 0$ do
            $A \leftarrow o$.getAncestor($l$)
            Parsimonize($o$,$A$)
            $l \leftarrow l - 1$
```

(b) Procedure of Specification. o.getAncestor(l) gives the ancestor of object o with l edges distance from it.

```
Generalization Stage
procedure Generalization
    Stack ← all objects in breadth first order
    while Stack is not empty do
        $o \leftarrow$ Stack.pop()
        $l \leftarrow o$.Height()
        while $l > 0$ do
            $D \leftarrow o$.getDecedents($l$)
            Parsimonize($o$,$D$)
            $l \leftarrow l - 1$
```

(c) Procedure of Generalization. o.getDecedents(l) gives all the decedents of object o with l edges distance from it.

```
Model Parsimonization
procedure Parsimonize($o$,$B$)
    for all term $x$ in the vocabulary do
        $P(x|\theta_B) \leftarrow \sum_{b_i \in B} \left( P(x|\theta_{b_i}) \prod_{\substack{b_j \in B \\ j \neq i}} (1 - P(x|\theta_{b_j})) \right)$
        repeat
            E-Step: $P[x \in X] \leftarrow P(x|\theta_o) . \frac{\alpha P(x|\tilde{\theta}_o)}{\alpha P(x|\tilde{\theta}_o) + (1-\alpha) P(x|\theta_B)}$
            M-Step: $P(x|\tilde{\theta}_o) \leftarrow \frac{P[x \in X]}{\sum_{x' \in X} P[x' \in X]}$
        until $\tilde{\theta_x}$ becomes stable
```

(d) EM procedure of model parsimonization.

Figure 5: Pseudo-codes for the procedure of estimating significant words language models.

We have conducted extensive experiments on the effectiveness of RSWLM for (pseudo-)relevance feedback problem and showed that its outperforms state-of-the-art methods [2, 3].

As one of the experiments, Table 1 presents the results of employing RSWLM as the feedback model as well as baseline methods on the task of PRF. We have also analysed the process of estimating RSWLM to understand how employing this idea enables the feedback system to control the contribution of feedback documents to prevents their specific or general terms affect the feedback model. For example, in the task of relevance feedback, it has been shown that there are some relevant documents that hurt the feedback performance by adding off-topic terms to the feedback model. These documents are called “poison pills” [7, 17].

Figure 4 shows how using RSWLM empowers the feedback system to deal with the poison pills. In this figure, the performance of the different systems in topic 374 on Robust04 dataset are illustrated. As can be seen, adding the seventh relevant document to the feedback set leads to a substantial decrement in the performance of the feedback in all the systems. The query is “Nobel prize winners" and the seventh document talks about the Nobel peace prize, but at the end, it has a discussion concerning Middle East issues, which contains some high frequent terms that are non-relevant to the query. However, RSWLM is able to distinguish this document as a poison pill and by reducing its contribution to the feedback model, i.e. a low value of $\lambda_{d_7,sw}$, it prevents the severe drop in the feedback performance.

4.3 Hierarchical Classification

In this section, we investigate the question: “*How to estimate SWLM of hierarchical entities? How effective is SWLM on classifying hierarchical entities and how accurate is the estimated models across time?*”

When we refer to the group of objects, we can think of a simple three layer hierarchical structure, where the individual objects take place in the lowest layer, groups are considered as nodes in the middle layer, and the set of all the objects in all the groups is deemed to be the supper node on the top layer. In this fashion, assuming the “general” model reflecting the characteristics of the top node in the hierarchy and “specific” model representing specific features of individual objects in the lowest layer, SWLM aims on estimating the latent models of intermediate group nodes so that these models are not affected by features from other layers.

With regards to this point of view, we extend the process of estimating SWLM to be applicable when there are more than three layers in the hierarchy. In this model, in order to extract SWLM for each object in the hierarchy, the parsimonization is done towards both all the ancestors in all the above layers as well as all the descendants of the object in the below layers.

More precisely, the estimation process is an iterative procedure that in each iteration, there are two main stages: a *Specification stage* and a *Generalization stage*. In the specification stage, the model of each object will be specified relative to its ancestors and in generalization stage, the model of each object will be generalized considering all its descendants. These two stages are repeated until all the estimated models of objects become stable.

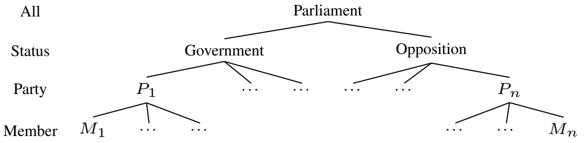

Figure 6: Hierarchical relations in parliament.

The pseudo-code of overall procedure of estimating SWLM is presented in Figure 5a. Figures 5d, 5b, and 5c present the pseudo-code of Specification Stage, Generalization Stage, and Model Parsimonization respectively.

Eliminating effect of other layers in estimating SWLM stands to reason that these models represent entities taking their position in the hierarchy into account. We have analysed the estimated SWLM of hierarchical entities and found out that they are highly separable, both vertically and horizontally [6]. Furthermore, we have employed SWLM on the task of hierarchical classification and observed that due to the horizontal and vertical separation property, the estimated models are precise, robust and transferable over time [5]

We used parliamentary hierarchy (Figure 6) for our experiments and tried to estimate models of parties and use them to classify members to their parties. In the parliament hierarchy, since members and parties can move in the hierarchy over different periods, cross period classification is notoriously challenging [11]. This is because we need to estimate models that are not affected by the positions of entities and are valid after changes.

As Table 2b show the performance of employing SWLM on party classification compare to SVM classifier. As can be seen, SVM performs well in terms of accuracy within period, but this performance is indebted to the separability of parties due to their status. Hence, changing the status in cross period experiments, using trained model on other periods fails to predict the party so the accuracies drop down. This is exactly the point that the strengths of our proposed method kicks in. Our proposed approach tackles the problem of having non-stable models when the composition of parliament evolves during the time, by capturing the essence of language models of entities at aggregate levels. In other words, rather than using adaptive machine learning to update the model to changes in the data stream, SWLM builds abstract models that transfer well over different time periods.

5. REST OF THE JOURNEY

There are some interesting directions that we envision to be followed in this research. For example, applying the model in other applications like “Authorship Attribution or Profiling”, in which the

Table 2: Results on the task of party classification.

(**a**) Accuracy of SVM classifier

	Period	**Test**			
		2006-10	2010-12	2012-14	All
Train	2006-10	47.56	29.22	26.84	-
	2010-12	29.87	40.90	35.57	-
	2012-14	31.09	30.51	44.96	-
	All	-	-	-	39.18

(**b**) Accuracy of classifier uses SWLM

	Period	**Test**			
		2006-10	2010-12	2012-14	All
Train	2006-10	44.51	46.10	43.62	-
	2010-12	40.85	40.25	39.59	-
	2012-14	40.24	38.96	42.28	-
	All	-	-	-	49.94

goal is to extract permanent latent model of authorship by eliminating the effect of insignificant features, like the topic of documents.

Beside applying the model in other area, here we want to refer to some of new directions and discuss difficulties and challenges ahead. The first direction would to address **RQ3**: “*How can SWLM be used as an analytical tool, which gives key insights into the characteristics of the data?*”

Generally speaking, SWLM uses the information from mutual relations of documents to decompose the data into different components. This provides new analytical handles to investigate and better understand the data. For example, with regards to this ability, analysing the estimated SWLM for hierarchical entities, we have studied the necessity of two-dimensional separation in the hierarchical models for hierarchical classification and showed how separation improves the accuracy of the decisions made by classifiers [6].

As another interesting problem, in the feedback application, estimating SWLM can be seen as decomposing the score or ‘retrieval status value’ of documents into three components: relevant, specific, and general. This allows a better understanding of the concept of relevance in information retrieval and provides more accurate unsupervised estimates of the probability of relevance.

Since SWLM decomposes the way each document is related to other documents in a group, i.e contribution of each of specific model, general model, and significant words model in the document, it could be used as an analytical tool for investigating the mutual relations in the group as well as the share of each individual document in the significant word model. As a specific problem in this direction, we are interested in using such detailed information to study the effect of retrieved documents,based on being relevant, specific or general, on users behavior in search environments. For example, in query reformulation, it is interesting to determine that each term that the user adds to its query after the first run of retrieval is a specific term, a general term, or one of the significant terms.

On the other hand, we named our model, significant “words” language model in honor of Luhn, however, it could be employed in non-textual environments, since in general the idea is to extract the significant “features” representing the shared essence of a group of objects. Considering this fact, we can follow our four research question: **RQ4**: “*How to apply the idea of SWLM in other environments for example on the output of embedding methods to improve the final representations?*”

Embedding methods like WordToVec, are becoming popular and there are similarities between challenges in the embedding methods and the problem we are addressing in our research. For example, one of the existing problems is given words embeddings, how to embed sentences and then embed documents and also users. From our point of view, this question can be translated as “how to estimate a model for a group of objects having individual models”, which SWLM tries to confront with. Besides, employing SWLM as an analytical tool in this framework is desirable in order to understand the semantics behind these embeddings.

However, the challenge is how to apply the idea of SWLM in other frameworks, like neural networks to learn embedding for multiple entities. One option would be applying the model on the output of embedding methods. But, the difficulty is how to deal with the fact that the vectors are not probabilistic distributions while SWLM is based on probabilities. The naive solutions like vectors normalization are not necessarily applicable since they may destroy the meaning behind the vectors. One other approach would be designing a new neural network, so that the hierarchical structure among words, sentences, documents, and so on are taken into account and the network controls the insignificant information (in accordance with SWLM idea) during the training phase.

The aforementioned are the main research questions that we are going to investigate and address in the rest of our research.

REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[2] M. Dehghani, S. Abnar, and J. Kamps. The healing power of poison: helpful non-relevant documents in feedback. 2016. Submitted to CIKM ’16.

[3] M. Dehghani, H. Azarbonyad, J. Kamps, D. Hiemstra, and M. Marx. Luhn revisited: Significant words language models. 2016. Submitted to CIKM ’16.

[4] M. Dehghani, H. Azarbonyad, J. Kamps, and M. Marx. Generalized group profiling for content customization. In *CHIIR ’16*, pages 245–248, 2016.

[5] M. Dehghani, H. Azarbonyad, J. Kamps, and M. Marx. Two-way parsimonious classification models for hierarchical texts. In *Proceedings of CLEF’16*, 2016.

[6] M. Dehghani, H. Azarbonyad, J. Kamps, and M. Marx. On horizontal and vertical separation in hierarchical text classification. 2016. Submitted to ICTIR ’16.

[7] D. Harman and C. Buckley. Overview of the reliable information access workshop. *Inf. Retr.*, 12(6):615–641, 2009.

[8] S. H. Hashemi, M. Dehghani, and J. Kamps. Parsimonious user and group profiling in venue recommendation. In *TREC 2015*. NIST, 2015.

[9] B. He and I. Ounis. Studying query expansion effectiveness. In *Proceedings of ECIR ’09*, pages 611–619, 2009.

[10] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings of SIGIR’04*, pages 178–185, 2004.

[11] G. Hirst, Y. Riabinin, J. Graham, and M. Boizot-Roche. Text to ideology or text to party status? *From Text to Political Positions: Text analysis across disciplines*, 55:93–15, 2014.

[12] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of SIGIR ’01*, pages 111–119, 2001.

[13] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of SIGIR ’01*, pages 120–127, 2001.

[14] H. P. Luhn. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2(2):159–165, 1958.

[15] Y. Lv and C. Zhai. Revisiting the divergence minimization feedback model. In *CIKM ’14*, pages 1863–1866, 2014.

[16] T. Tao and C. Zhai. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *Proceedings of SIGIR ’06*, pages 162–169, 2006.

[17] E. Terra and R. Warren. Poison pills: Harmful relevant documents in feedback. In *Proceedings of CIKM ’05*, pages 319–320, 2005.

[18] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of CIKM ’01*, pages 403–410, 2001.