Pak. J. Engg. Appl. Sci. Vol. 23 July, 2018 (p. 77–85)

Finding Topics in Urdu: A Study of Applicability of Document Clustering on Urdu Language

Toqeer Ehsan*, H. M. Shahzad Asif

Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan

∗ **Corresponding Author:** Email: toqeer.ehsan@gmail.com

Abstract

In this research, we present the results of a study conducted to ascertain the applicability of document clustering techniques on Urdu language corpus. This study, which is first of its kind, employs a fully probabilistic Bayesian method, Latent Dirichlet Allocation, for clustering Urdu language corpus by using the features collected from the documents. Results obtained are compared with those obtained from a simplistic classification technique. Analysis of the results shows that supervised and unsupervised techniques for grouping documents perform reasonably well on this corpus. Results further indicate that Urdu document clustering technique outperforms document classification technique in some cases with an accuracy of above 90%.

Keywords: Unsupervised, LDA, Urdu, Text Clustering, Classification, Corpus, Stemming, Naïve Bayes

1. Introduction

Topics modeling could be supervised as well as unsupervised. In the supervised topic modeling, predefined labels are already available which are used to train the system. On the other hand, unsupervised method is a bit trickier. In the unsupervised modeling no labels are given so that the system needs to compute the assignments of topics by itself. Unsupervised topic modeling considers the documents as the mixture of latent topics. This work mainly focuses on unsupervised Urdu topic modeling based on latent Dirichlet allocation (LDA). If we consider a document about computer science then there must be a set which contains most frequently used words in the field. But it does not mean that all the word in this document only belong to this topic. There must be some other words as well that may belong to other topics as well like engineering, electronics etc., but these topics are hidden. A topic model finds a certain number of different topics from a set of documents. A document may not be associated with a single topic. It has potential to have associations with other topics as well. The topic model computes these proportions of the topics within documents. Topic modeling has been applied to numerous domains including text clustering, document tagging, movie genre identification, online course recommendation, stock market price predictions.

Ali and Ijaz (2009) performed supervised Urdu text classification using Naïve Bayes classifier and Support Vector Machines (SVM) [1]. They have performed multi-level preprocessing on training data before applying the models. Tokenization has been done on the basis of the lexicon. They have eliminated the stop words and diacritics from the text. Furthermore, they performed the affixes basis stemming as Urdu language is a morphologically rich which contains many surface forms against a single root. They have compared the classification results of different data sets and concluded that classifiers perform well on Urdu without stop words and diacritics. They have further concluded that stemming does not improve the classification results that much. Similarly, [2] has implemented a supervised classification framework for Arabic text. Their framework is based on Latent Dirichlet Allocation (LDA) and Support Vector Machines (SVM). They have performed multiple tasks for data normalization including transformation of Arabic characters to simplified characters, elimination of diacritics, elimination of non-Arabic words, elimination of functional words and stemming. Their corpus contains the text from nine different domains. F_1 (micro averaging and macro averaging) measures are calculated from classification results. Their results outperformed Support Vector Machines (SVM), Naïve Bayes and k-Nearest Neighbour classifiers for Arabic text.

Latent Dirichlet allocation (LDA) based topic modeling has been applied to various domains including text categorization, document tagging, stock market price and movie genre prediction. LDA based topic modeling are applied to predict the tags for document abstracts in [3]. They used a set of 200 abstracts from four

different topics. They have concluded that LDA with Gibbs sampling outperforms the CVB0 sampling algorithm. Similarly, [4] has proposed a method to perform topic modeling for short texts. They calculated topic-word matrix to measure the topic similarities. In their proposed method, they have used the distance matric from KNN algorithm to compute topic similarities. LDA topic modeling has also been applied to recommend the online courses for students in [5]. Their system has used the syllabus and contents of the courses to perform the predictions. Hollywood movie genres have been predicted by applying LDA topic modeling in [6]. Their data set consists of movie scripts in textual form. They have collected the scripts from Hollywood movies which are released from 1935 to 2015. They have used Mallet (machine learning for language toolkit) for training and testing. They have also compared the results with other classification algorithms and have shown the improvement of LDA based topic modeling. [7] have introduced a new method to predict the stock market behaviour by using social media sentiment as a feature. Their method outperformed the model based on historical data by 6.07%.

LDA topic modeling has been used to improve the dictionary based sentiment propagation as described in [8]. A context-aware method has been proposed by generating the specific topics against the specific contexts. The experiments have been performed on Chinese ConceptNet and the context-aware method performs better than context insensitive sentiment propagation. A variation of LDA model has been proposed by using the word-embeddings in [9]. A large corpus has been used to compute the word vectors and they are used to classify small text. This embedding-based topic modeling (ETM) performs better on smaller text documents. Similarly, another method has been proposed to filter out the noise in short texts as presented in [10]. They perform common semantics topic modeling (CSTM) by introducing a new common topic for each text to identify noise. The experiments perform better than existing short text topic modeling. LDA topic modeling can be supervised [11] as well as unsupervised but [12] have proposed a new semi-supervised method for text classification and have compared the results with existing techniques.

Urdu is a morphologically rich and comparatively low resourced language. Performing the unsupervised topic modeling for Urdu for a small to medium sized corpus would lead to data scarcity. The preprocessing of the corpus is crucial to get the compatible language representation. Part of speech tagging has been used to extract the content words in the corpus. The results of the unsupervised topic modeling have been compared with the supervised text classifier. Section 2 discusses the inference methodology used in LDA topic modeling. In section 3, we give the details of corpus used for the experiments. This section describes the Urdu text representation and preprocessing steps. Section 4 explains the naïve Bayes classifier. Section 5 and 6 present the experimental setup and the results of topic modeling and comparison with Naïve Bayes classifier.

2. Inference Methodology

Latent Dirichlet allocation (LDA) is a generative graphical model to identify topics from a given set of documents. LDA is a Bayesian network model which is based on unsupervised learning paradigm. LDA was first presented by David Blei, Andrew Ng and Michael Jordan in 2003 [13]. The model considers each document as a mixture of topics and determines word-topic and document-topic matrices. Word-topic matrix contains the information about words belonging to topics with respect to probability. If we have N number of words in all the documents i.e. $\{w_1, w_2, \ldots, w_n\}$ and K topics i.e. $\{t_1, t_2, \ldots, t_k\}$ then this matrix shows the proportion of *n-th* word with *k-th* topic. Document-topic matrix contains the information about the association of documents with topics. If there are D number of documents i.e. $\{d_1, d_2, \ldots d_d\}$ and K topics $\{t_1, t_2, \ldots t_k\}$ then this matrix gives the proportion of *d-th* document with *k-th* topic. LDA uses Dirichlet distribution as the prior for the multinomial distribution because Dirichlet distribution is a conjugate prior of multinomial. It uses Gibbs sampling to compute word samples among the distribution of words. Since LDA deals with the words and there occurrences so it is also called a bag of words model. The model can be represented by the following Fig.1.

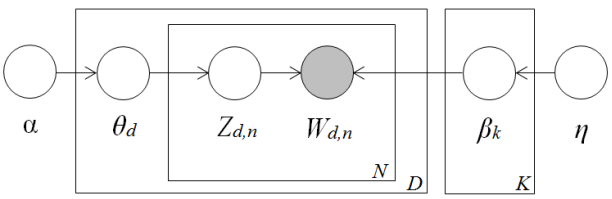

Fig. 1: Graphical model of LDA

Each node in Fig. 1 is a random variable. Directions of the edges show the dependencies among the variables. Shaded node is observed

variable and unshaded nodes are unobserved (latent) variables. Details of these variables are as follows:

$\boldsymbol{\theta_d}$ – is the per document topic proportions, one value for each document.

$\boldsymbol{Z_{d,n}}$ – per word topic assignment. It is depended on θ_d topic assignment to *n-th* word in *d-th* document.

$\boldsymbol{W_{d,n}}$ – it is the *n-th* word in *d-th* document. It is the only observed random variable in the whole model. It depends on $Z_{d,n}$ and β_k.

$\boldsymbol{\beta_k}$ – K-topics: Each β is the distribution over terms. It comes from Dirichlet distribution.

$\boldsymbol{\alpha}$ – is the proporti ons parameter and controls the mean shape and sparsity of θ.

$\boldsymbol{\eta}$ – is the topic parameter.

LDA is an unsupervised text clustering model, unlike other clustering techniques; it provides the labels of the clusters. It gives the list of most frequent words for each topic from which the topic names can be inferred. Joint probability distribution (JPD) of all the observed and latent variables is defined by the Eq.1.

$$P(\beta, \theta, \boldsymbol{Z}, \boldsymbol{W}) = \prod_{i=1}^{K} P(\beta_i|\eta) \times \prod_{d=1}^{D} P(\theta_d|\alpha) \times \prod_{n=1}^{N} P(Z_{d,n}|\theta_d) P(W_{d,n}|\beta_{1:K}, Z_{d,n}) \quad (1)$$

Dirichlet distribution is the multivariate probability distribution which belongs to the exponential family of distributions. It is the conjugate prior of multinomial distribution. Probability density function without coefficient can be written as shown by Eq.2.

$$P(x \mid \alpha) = \prod_{i=1}^{K} x_i^{\alpha_i - 1} \quad (2)$$

Gibbs sampling is a Monte Carlo Markov chain (MCMC) technique to find the series of observations from multivariate probability distributions [14]. It is a randomized algorithm which is normally used for Bayesian inference. Due to randomization it produces different samples each time when it is run. Gibbs sampler forms a Markov chain on the basis of sequence of observed samples. Now, suppose we want to compute *K* number of samples of $X = (x_1, x_2, \ldots x_n)$ where joint distribution is $P(x_1, x_2, \ldots x_n)$. At *i-th* iteration the sample is $X_i = (x_i1, x_i2, \ldots x_in)$. The algorithm works as follows:

1. Start with some initial value X_0.
2. Let's next sample is $(i+1)$. To compute next sample $x_{ji}+1$ the distribution is P (xj | $x1_{i+1}, \ldots, x_{i+1j-1}, x_{ij+1}, \ldots, x_{in}$). Note that x_j is not included in the conditional probability and secondly, Gibbs samples use the latest updated samples to compute the new samples. Here, by computing x_j , 1 to *J* samples are already computed but other values are being used of previous iteration.
3. Repeat second step *K* times.

Conditional probability given all other variables is shown by Eq.3 as follows:

$$P(x_j | x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n) = \frac{P(x_1, \ldots, x_n)}{P(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)} \propto P(x_1, \ldots, x_n) \quad (3)$$

Gibbs sampling is used to compute the terms for topics. Above algorithm can be mapped on the textual data. LDA uses a *1*x*N* vector that contains the indices of the words and there are total *N* numbers of words. The other vector contains all the words on document indices. If we want to perform topic modeling for *K* topics and LDA produces at most *T* terms for each topic then model performs Gibbs sampling that computes *T* samples for each topic.

3. Corpus Design and Collection

Urdu is a highly spoken South Asian language which is written in Arabic script. It has a number of differentiating features as compared to English. Firstly, Urdu has a strong case marking system [15] [16]. It uses different clitics to mark the cases which are also referred as postpositions. Urdu also has prepositions but there are very few examples in the corpus. Secondly, Urdu is highly rich in morphology as compared to English [17] [18]. According to [18], an Urdu verb can have more than 50 surface forms. It also has different surface forms for causatives and double causatives. Thirdly, Urdu has a structure where nouns, adjectives or quantifiers give verbal sense when following a light verb. This phenomenon is referred as complex predicate structure [19]. Urdu frequently uses auxiliary verbs which make its verbal structure way complex from English. To perform topic modeling on Urdu text, preprocessing is an important task due to its differences with English. For this purpose part of speech tagged documents have been used [20] [21] [22]. POS tagging is very helpful while performing preprocessing which is discussed in coming section.

The text in our corpus belongs to five different genres with multiple text documents. Each

document contains three hundred words on average. This count represents the number of tokens before performing any kind of preprocessing tasks. The text is written in Unicode based Arabic script. Table 1 shows the statistics of the corpus and bar chart in the Fig. 2 illustrates the token to term ratio for each text class. The corpus is annotated with part of speech tags. The word delimiter is space character while for part of speech tags we have used slash (/) as a delimiter. Following natural language processing (NLP) tasks have been performed to process the corpus.

3.1. POS Tagging

Part of speech (POS) tagging has been applied to the experimental data to identify the tokens and their importance in the model. Most important terms are common nouns and proper nouns i.e. قرض (debt), شعبہ (department) etc. Another example, انتظام (manage) is a noun in Urdu but in English it is a verb. This kind of strcuture is called complex predicate construction in Urdu . It does not only happen with nouns but also with adjectives and quantifiers. Urdu part of speech tag set is more flat in nature and does not mark them separately. The tag set just treats them as common nouns, adjectives or quantifiers. POS tag set has two types of verbs, infinitive and finite verbs i.e. ہونے (to be), آنے (to come) etc. and کہا (said), کی (did) etc. respectively Therefore, in our model commons nouns, proper nouns and adjectives are considered.

3.2. Stopwords/Punctuation Elimination

All the stop words that are only helpful in grammatical structure are removed as they have not much role in topic modeling [1]. Some Urdu stop words are: تھا , ان , ہے , کو, کے , پر etc. Punctuation marks are also removed like: ; ,: ,”, ؟ , / ,(,) etc. Diacritic symbols are optional in Urdu therefore they have been removed in order to avoid any kind of ambiguities among tokens.

3.3. Stemming

Stemming is a process of removing affixes from the tokens. Normally there are two types of affixes, prefixes and postfixes. Prefixes appear at the start of the stem and postfixes (suffixes) appear at the end. For example in the token بےشک , بے is the prefix of the word. Similarly, in the token رضامند the portion مند is the suffix. A stemmer simply trims the tokens on the basis of affixes. It usually uses the lists of affixes to perform stemming. In the Urdu text classification stemming does not play much important role as suggested in [1]. However, we have performed the experiment on the stemmed data as well. The topic modeling results are compared after training on different datasets.

3.4. Dataset

We have chosen small-medium sized Urdu tagged corpus to train the LDA model. Initially, five topics are selected to process the model. Table 1 shows the statistics of terms and tokens with respect to topics.

Table 1: Urdu corpus statistics

Topics	Documents	Tokens	Terms
Sports	29	9,815	2,942
Health	29	9,675	3,267
Culture	28	8,530	2,846
Entertainment	14	4,923	1,699
Religion	29	9,746	2,673
Total	129	42,689	13,427

After preprocessing, useful terms are obtained to perform the execution of the algorithms. Fig. 2 demonstrates the term-token ratios for each class. The class of Entertainment contains a lower number of terms so the results for this category may be crucial.

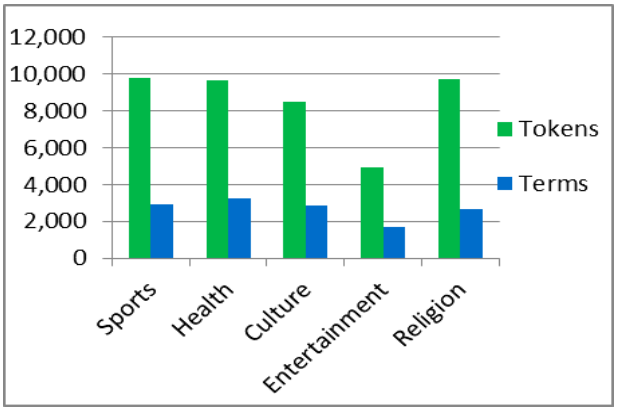

Fig. 2: Tokens and terms of each topic in corpus

4. Bayesian Classifier

Naïve Bayes classifier is a probabilistic classification method with independence assumption. All the features are supposed to be independent of each other. It is an efficient classification algorithm due to its simplicity and works well for text classification. It is based on Bayes rule as follows:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (4)$$

$P(A|B)$ = Posterior Probability

P(A) = Prior Probability of the class
P(B|A) = Likelihood which is the probability of predictor given class
P(B) = Prior probability of predictor

Key feature for the classification is the term frequency in the documents. As the algorithm assumes the independence among the features so each term frequency is computed separately. Now, Eq.4 can also be written as:

$$P(Class|Doc) = \frac{P(Doc|Class)P(Class)}{P(Doc)} \quad (5)$$

Or

$$P(Class|Doc) = \frac{[\prod_j^n (Term_j|\ Class_i)\ ] \times P(Class_i)}{\prod_k^n (Term_k)} \quad (6)$$

Factor in the denominator remains constant for each class so it can be ignored. Now if *y* is the class and *x* is the term and there are *n* terms then we can write Eq.6 as:

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^n P\ (x_i \mid y) \quad (7)$$

And

$$\hat{y} = arg\ max_y\ P(y) \prod_{i=1}^n P\ (x_i \mid y \quad (8)$$

The section 6 describes the results of Naïve Bayes classifier on the supervised data and the comparison with LDA topic model.

5. Experimental Setup

To perform the topic modeling for Urdu, Mallet [23] has been employed for training and testing on our text corpus. It provides a command line interface to import data, train topics and inference. It implements LDA model based on Gibbs sampling. Mallet can import a single input file as well as a complete directory containing the files to train the topics. For Urdu topic modeling all the documents are given to the model as input. The model works in two steps; at first step it imports the corpus by taking the corpus path and a couple of parameters for data representation which are:

- *keep-sequence:* when this flag is on, the model reads the text as the sequence of words rather than vector representations.
- *remove-stopwords:* This flag enforces the elimination of stop words from the training corpus but it works only for English. For Urdu we have performed the preprocessing which removes the stop words from our corpus.

At the second step the topics are trained and this step also sets few parameters for topic modeling which are:

- *num-topics:* This parameter takes the value for total number of topic trained on the corpus. In our case the number of topics are five.
- *optimize-interval:* This option sets the interval for the optimization of hyperparameters. We have set the optimization after every 20 iterations.
- *output-state:* This option outputs a file with topic state which shows the topic assignement for each token.
- *output-topic-keys:* This option outputs a file with topic keys by providing the top key tokens for each topic.
- *output-doc-topics:* This option outputs a file which shows the proportion of each document in the training corpus to all the topics.

As we have already categorized the data into topics therefore, it has been intuitive to analyze the accuracy of the model. For Bayesian classification, we have programmed the classifier as discussed in section 4. The corpus has been divided into the train and test sets. After training the topics the classifier has been evaluated for test documents. The accuracy of the classifier has been compared with the accuracy of the unsupervised topic modeling as given in the section 6.4.

6. Results and Discussion

LDA model has been train for five topics and the results are reasonable. The accuracy has been calculated by dividing the correctly clustered documents with the total number of documents for same topic. Following sections describe the results and comparison with supervised classifier.

6.1. Topic Keys

Topic keys for all the trained topics are given in Table 2. We can understand the class after examining the terms for each topic. These keys are sorted in reverse order. For example the word کھیل has highest importance and frequency for Topic 0. Similar procedure is for all other topics and terms.

6.2. Topic Accuracy

Fig. 3 exhibits the performance of the model for each category. It can be assessed that the discrete topics have higher accuracy as compared to most common and generic topics like culture and entertainment. The topic accuracy has been computed by dividing the correctly clustered documents with total number of documents. Next section discusses the document topic association and behavior of the outcomes by presenting the f-measures for three experiments.

Table 2: Urdu topic keys from LDA model

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
کھیل (sport)	مریض (patient)	جنگ (war)	یہاں (here)	حضرت (majesty)
کرکٹ (cricket)	استعمال (use)	فوج (military)	زبان (language)	جب (when)
کھلاڑی (player)	خون (blood)	خلاف (against)	شہر (city)	دن (day)
ٹیم (team)	جسم (body)	بادشاہ (king)	نام (name)	ہاتھ (hand)
جب (when)	درد (pain)	شہر (city)	سمندر (ocean)	وقت (time)
وقت (time)	کام (work)	وہاں (there)	سفر (voyage)	لوگ (people)
کھلاڑیوں (players)	پانی (water)	ساتھ (beside)	بڑے (big)	طرح (like)
بعد (after)	علامات (symptoms)	زندگی (life)	جہاز (ship)	پاس (have)
دنیا (world)	علاج (cure)	انگریز (englishman)	میل (mile)	کام (work)
صرف (only)	بعد (after)	تاریخ (history)	بعد (after)	بن (son of)
میچ (match)	مرض (disease)	طور (in a way)	ہر (all)	لوگوں (people)
پاکستان (Pakistan)	تحقیق (research)	حکومت (government)	طرف (side)	بعد (after)
گیند (ball)	ذہنی (psychological)	آزادی (freedom)	ساتھ (beside)	بات (talk)
دن (day)	طور (in a way)	بغاوت (rebellion)	قریب (near)	گھر (home)
سال (year)	ہر (all)	بڑی (big)	پہل (first)	ساتھ (beside)
میدان (playground)	لہسن (garlic)	قتل (murder)	ملک (country)	اللہ (God)
طرح (like)	ذیابیطس (diabetes)	اب (now)	مسجد (mosque)	شروع (start)
پاکستانی (Pakistani)	مقدار (quantity)	حکیم (wise)	تاریخ (history)	طرف (side)
حاصل (obtain)	افراد (people)	مالک (owner)	واقع (situated)	نام (name)

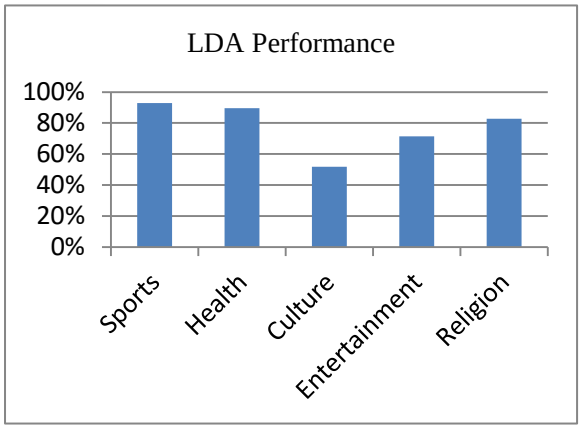

Fig. 3: Topic model performance for each class

6.3. Topic Proportions

Topic proportions table is an error matrix which shows the overlapping topics. We have computed precision, recall and F1-score against three experiments performed on three different datasets. First dataset is prepared by removing stop words, second dataset is prepared by performing stemming on the first dataset, and third dataset uses POS tags to extract the content words for topic modeling.

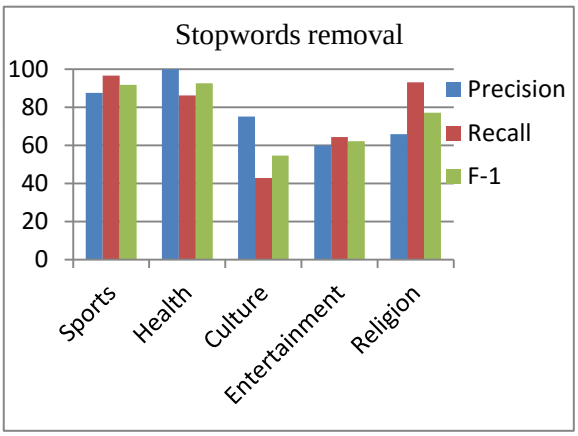

Fig. 4: Topic model performance for each class

Fig. 4 shows the results of topic modeling trained after removing stop words from the training data. A lexicon containing Urdu stop words has been used to remove the stop words from the dataset. The f-scores for Sports, Health and Religion are quite promising. Fig. 5 shows the f-

measures after performing stemming on the first dataset.

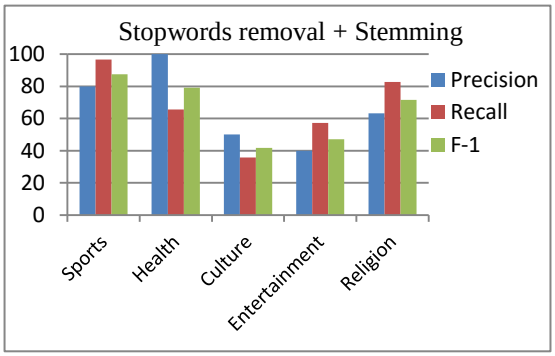

Fig. 5: Topic model performance for each class

The process of the stemming replaces the words with their stems after removing the affixes (prefixes and suffixes). Stemming is helpful to reduce the data sparsity. An existing Urdu stemmer[1] has been used to prepare third dataset. The stemmed dataset has not shown any improvement in the f-measures however, it decreases the f-score for all the classes. The POS tagged dataset gives the highest f-measures for all text classes as shown in Fig. 6.

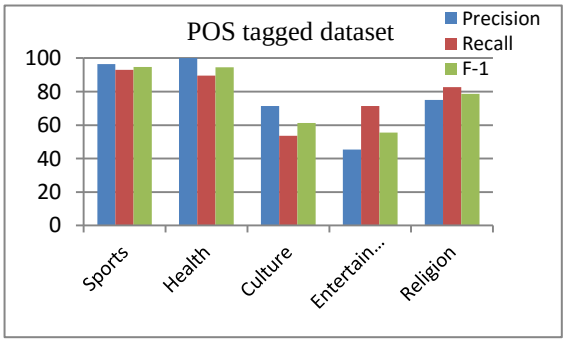

Fig. 6: Topic model performance for each class

Table 3 shows the document-topic proportion after training the model on already classified Urdu text corpus.

Table 3: Urdu topic proportions

Category	Topi 0	Topic 1	Topic 2	Topic 3	Topic 4
Sports	93%	0%	3.5%	0%	3.5%
Health	0%	6.9%	0%	89.6%	3.5%
Culture	3.5%	17.2%	27.5%	0%	51.7%
Entertainment	0%	7.1%	71.4%	0%	21.5%
Religion	0%	82.7	13.8%	0%	3.5%

[1] CLE Urdu Stemmer, URL:http://cle.org.pk/clestore/urdustemmer.htm

Three topics, Sports with 93%, Health with 89.6% and Religion with 82.7% accuracy seem to work fine. Other two categories, Culture and Entertainment are bit overlapping. There could be three reasons for this similarity. First, model was not working properly. Second, we have observed before that entertainment group was having less number of terms as compared to other classes. Third, these two topics are connected to each other naturally. To answer these question lets apply supervised text classification algorithm to compare the results with LDA model. We have selected simple and efficient text classifier named Naïve Bayes classifier. Next section gives the comparison of Naïve Bayes classifier with LDA model.

6.4. LDA vs Naïve Bayes

Naïve Bayes classifier has been trained in the same Urdu text data. Almost 70% text for each class is used for training purposes and other 30% is used for evaluation. This ratio is selected because we are dealing with small to medium sizes text corpus. Fig. 7 shows the accuracy of the LDA model and Naïve Bayes classifier and the behavior of the consequences are quite similar. For the group with distinct text and terms like sports, LDA model even works better than Naïve Bayes. For the overlapping classes like culture and entertainment the accuracy of the supervised classifier is similar to the accuracy of the LDA topic model.

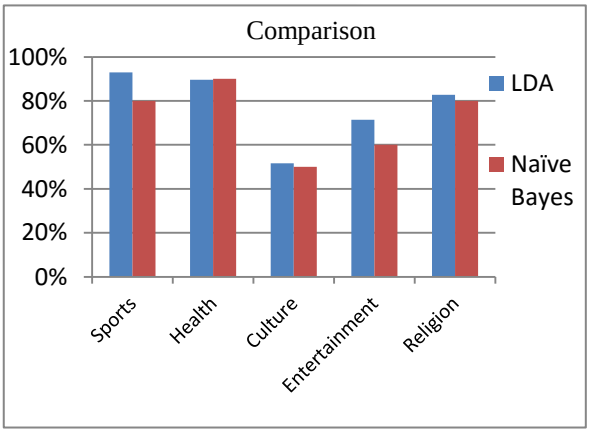

Fig. 7: Results comparison (LDA vs Naïve Bayes)

6.5. Computational Complexity

Naïve Bayes classifier is very efficient due to its simplicity and independence assumption. It treats all the classes independently and computes the associations of documents to one class at a time. On the other hand LDA computes the probability of each term to each topic which makes it less efficient. But a parallel implementation of

LDA model is available which makes it faster for large scale applications [24].

7. Conclusion

This work presents topic modeling based on latent Dirichlet allocation (LDA) for Urdu text. LDA is an unsupervised graphical Bayesian network model. It computes the latent topics in the text corpora. LDA uses Gibbs sampling algorithm to find the samples in the data. We have trained the model on a small-medium Urdu text dataset with predefined categories and the results are satisfactory. Topic modeling has been experimented on three different datasets, which are prepared after performing preprocessing tasks including stop words removal, stemming, and part of speech tagging. The stemming does not improve the results of the topic modeling for Urdu text. The part of speech tagged dataset uses the tags to extract the content words and it gives better performance. It produced few topics with overlapping assignments. The model has been further evaluated by training a supervised Naïve Bayes text classifier on the same data and the results are almost similar. Although, the unsupervised topic model results are better than the supervised Naïve Bayes text classifier. It is concluded that LDA works fine for small datasets as well as for Urdu language. LDA can also be trained on labeled documents by encoding the predefined labels to the model. Future intensions are to evaluate LDA model with other unsupervised text clustering techniques.

8. Acknowledgment

We are grateful to Center for Language Engineering (CLE), Al-Khawarizmi Institute of Computer Science (KICS), UET, Lahore for developing and providing Urdu tagged corpus, Urdu part of speech tagger and world list.

9. References

[1] Ali, A. R., & Ijaz, M. (2009). Urdu Text Classification. *7th International Conference on Frontiers of Information Technology (FIT).*

[2] Zrigui, M., Ayadi, R., Mars, M., & Maraoui, M. (2012). Arabic Text Classification Framework Based on Latent Dirichlet Allocation. *Journal of Computing and Information Technology, 20*, 125-140.

[3] Anupriya, P., & Karpagavalli, S. (2015). LDA Based Topic Modeling of Journal Abstracts. *International Conference on Advanced Computing and Communication Systems (ICACCS -2015).* Coimbatore, INDIA.

[4] Chen, Q., Yao, L., & Yang, J. (2016). Short text classification based on LDA topic model. *International Conference on Audio, Language and Image Processing (ICALIP).* Shanghai, China.

[5] Apaza, R., Cervantes, E. V., Quispe, L. C., & Luna, J. O. (2014). Online Courses Recommendation based on LDA. *SIMBig2014.*

[6] Chao, B., & Sirmorya, A. (2016). Automated Movie Genre Classification with LDA-based Topic Modeling. *International Journal of Computer Applications, 145*(13).

[7] Nguyen, T., & Shirai, K. (2015). Topic Modeling based Sentiment Analysis on Social Media for Stock Market Prediction. *ACL-IJCNLP 2015.* Beijing, China.

[8] Chou, P.-H., Tsai, R. T.-H., & Hsu, J. Y.-j. (2017). Context-aware sentiment propagation using LDA topic modeling on Chinese ConceptNet. *Soft Computing, 21*(11), 2911-2921.

[9] Qiang, J., Chen, P., Wang, T., & Wu, X. (2017). Topic modeling over short texts by incorporating word embeddings. *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 363-374).

[10] Li, X., Wang, Y., Zhang, A., Li, C., Chi, J., & Ouyang, J. (2018). Filtering out the noise in short text topic modeling. *Information Sciences, 456*, 83-96.

[11] Blei, D. M., & McAuliffe, J. D. (2008). Supervised topic models. *Advances in, 20*, 121-128.

[12] Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and LDA topic models. *Expert Systems with Applications, 80*, 83-93.

[13] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research, 03*, 993-1022.

[14] Casella, G., & George, E. I. (1992). Explaining the Gibbs Sampler. *The American Statistician, 46*(03), 167-174.

[15] Ahmad, T. (2009). *Spatial Expressions and Case in South Asian Languags, Ph.D. dissertation.* University of Konstanz, Germany.

[16] Butt, M. (2006). *Theories of Case.* Cambridge: Cambridge University Press.

[17] Abbas, Q. (2015, April 16). Morphologically rich Urdu grammar parsing using Earley Algorithm. *Natural Language Engineering, 21*(2), 1-36.

[18] Hussain, S. (2004). *Finite State Morphological Analyzer for Urdu, Unpublished MS thesis.* National University of Computer and Emerging Sciences, Pakistan.

[19] Butt, M., & Ramchand, G. (2001). Complex aspectual structure in Hindi/Urdu. *M. Liakata, B. Jensen, & D. Maillat, Eds*, 1-30.

[20] Ahmad, T., Urooj, S., Hussain, S., Mustafa, A., Parveen, R., Adeeba, F., . . . Butt, M. (2014). The CLE Urdu POS Tagset. *Language Resources and Evaluation Conference (LREC 14)*, (pp. 2920-2925). Reykjavik, Iceland.

[21] Sajjad, H. (2007). *Statistical Part of Speech Tagger for Urdu.* Lahore, Pakistan: MS thesis, National University of Computer and Emerging Sciences.

[22] Sajjad, H., & Schmid, H. (2009). Tagging Urdu Text with Parts of Speech: A Tagger Comparison. *12th Conference of the European Chapter of the Association for Computational Linguistics (EACL-09).*

[23] McCallum, & Kachites, A. (2002). MALLET: A Machine Learning for Language Toolkit. *http://mallet.cs.umass.edu.*

[24] Wang, Y., Bai, W., Stanton, M., Chen, W.-Y., & Chang, E. Y. (2009). PLDA: Parallel Latent Dirichlet Allocation for Large-Scale Applications. *5th International Conference on Algorithmic Aspects in*, (pp. 301-314). Heidelberg.

[25] Zhang, H. (2004). The Optimality of Naive Bayes. *17th International FLAIRS Conference.* Florida, USA.

[26] *Matlab Topic Modeling Toolbox 1.4*. (n.d.). Retrieved from http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm

[27] Hastings, W. (1970). Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika, 57*(1), 97-109.

[28] Newman, D., Asuncion, A., Smyth, P., & Welling, M. (2009). Distributed Algorithms for Topic Models. *The Journal of Machine Learning Research, 10*, 1801-1828.

[29] Muaz, A., Ali, A., & Hussain, S. (2009). Analysis and Development of Urdu POS Tagged Corpus. *7th Workshop on Asian Language Resources, IJCNLP'09.* Suntec City, Singapore.

[30] Ahmad, T., Urooj, S., Hussain, S., Mustafa, A., Parveen, R., Adeeba, F., . . . Butt, M. (2014). The CLE Urdu POS Tagset. *Language Resources and Evaluation Conference (LREC 14).* Reykjavik, Iceland.

[31] Wang, Y., Bai, W., Stanton, M., Chen, W.-Y., & Chang, E. Y. (2009). PLDA: Parallel Latent Dirichlet Allocation for Large-Scale Applications. *5th International Conference on Algorithmic Aspects in.* Heidelberg.

[32] Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and LDA topic models. *Expert Systems With Applications, 80*, 83-93.