Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2019”

Moscow, May 29—June 1, 2019

APPLYING AN AUTOMATIC FTD CLASSIFIER TO THE ANNOTATION OF THE GICR CORPUS

Bulygin M. V. (bulyginmv1996@gmail.com)
Radboud University, Nijmegen, Netherlands

Sharoff S. A. (s.sharoff@leeds.ac.uk)
Russian State University of Humanities, Moscow, Russia;
Leeds University, Leeds, UK

This paper addresses the task of automatic genre classification for Russian within the Functional Text Dimensions (FTD) framework. Our aim in this study was to build the optimate FTD classification model to annotate web texts from the GICR corpus. For training data, we used an extended GICR dataset. We used the Support Vector Machine method with linear kernel for classification and converted training data to lower case to increase accuracy. During our research we experimented with several classification parameters, such as types of features, C-value and feature filtering to determine the best option for the classification model of the GICR dataset. The resulting model was able to achieve satisfactory classification accuracy and was used for GICR annotation. We also looked at the most significant features for each FTD in our best performing model and compared them to the most frequent words in which these features occur. Finally, we applied our model to segments of the GICR and looked at the FTD components in these segments.

Key words: Functional Text Dimensions, genre classification, feature selection, Web corpora annotation

ПРИМЕНЕНИЕ АВТОМАТИЧЕСКОГО FTD КЛАССИФИКАТОРА ДЛЯ АННОТАЦИИ КОРПУСА ГИКРЯ

Булыгин М. В. (bulyginmv1996@gmail.com)
Университет Радбауд, Неймеген, Нидерланды

Шаров С. А. (s.sharoff@leeds.ac.uk)
Российский государственный гуманитарный университет,
Москва, Россия; Университет Лидса, Лидс, Великобритания

1. Introduction

Language corpora evolved dramatically since the introduction of the first corpora. We started with a 1-million-word corpus that was collected manually and had restricted annotation, and nowadays we have massive language corpora that usually contain at least over 100 million words and have different sophisticated technics for corpus annotation, like POS-tagger, morphology and syntax analyzer and parser, etc. In some cases, we might even see megacorpora, with billions of words in them. For example, General Internet-Corpus of Russian (GICR) [Piperski et al., 2013] represents a variety of texts from the Russian web and is comprised of 20 billion words. It is obvious that such corpora cannot be constructed and annotated manually. The solution of this problem lies in the automatization of the process. Hence the various machine learning techniques are introduced to this field.

In this study we are researching the field of automatic genre classification for Russian. The importance of such classification is immense, especially for web corpora, where texts are collected by a web crawler, and the main purpose of the texts is not always clear.

Accurate genre classification can ease a user's navigation through the corpus, and allow scientists to research the difference in language use in various language subclasses.

Another problem that occurs when working with big web corpora is choosing the suitable genre classification system. Here, we are looking for the system that would incorporate in itself the balance between distinguishing ability and an adequate amount of genre labels. From the perspective of theoretical text typology, which tries to cover all the possible text variations, the number of genres in a language is extremely high. For example, [Gorlach, 2004] lists around 2,100 genres for English and [Adamzik, 1995] differs over 4,000 genres for German. Such classification systems satisfy the theoretical necessity of describing all types of texts, but are absolutely impractical. The classification system for corpus annotation needs to have a reasonable number of genre labels, in order to collect an adequate sub-corpus for each genre and to be convenient for the users [Sharoff, 2018]. The classification system for a web corpus should also reflect the diversity of Internet texts. Web texts have a strong tendency for hybridism between genres and new types of texts appear on the web all the time [Santini et al., 2010]. Ideally, our classification system should be able to consider and represent all of that information.

In our study we adopt the Functional Text Dimension (FTD) [Sharoff, 2018] approach for genre classification. We chose this classification framework because it has great coverage ability comparable with that of a long list of genre systems, while maintaining a relatively short list of genre labels. This result is achieved through the introduction of functional dimensions of text instead of discrete genre labels. In total there are 18 functional dimensions, which represent different language functions. In the majority of genre systems a text is believed to belong to only one specific genre, but in the FTD framework texts are described in each functional dimension independently. Thus, one text can score positive results in several dimensions simultaneously. The FTD classification system was designed to describe any text found on the web. Since each FTD represents a specific language function, we can use their combination to describe text hybridism and possible new genres, which can often be found in web texts.

To build a classifier we need a reliable manually annotated training corpus. In the FTD framework annotators are presented with a key question for each functional dimension. Depending on their answer, a text is classified as strongly (scored as 2), partially (scored as 1) or not at all (is a default score 0) belonging to the functional dimension.

2. Data

One part of our training data is a piece of the GICR corpus [Piperski et al., 2013], which was collected and annotated in terms of FTD in [Sharoff, 2018]. This corpus was later extended with around 500 new annotated texts from GICR by Serge Sharoff. In this study, we are using the resulting corpus as our training data.

The GICR corpus consists of texts from a variety of genres from the Russian web, such as blogs, news sites, social media etc. The annotated corpus was split on training and testing data. The split was approximately 90% of texts on training to 10% of texts on testing (see **Table 1**).

Table 1: Size and composition of the GICR dataset

Data set	Documents	Words
Training data	1,800	2,249,818
Testing data	140	163,923
Total	1,940	2,413,741

Manual corpus annotation is an essential, but extremely time-consuming task. Because it demands a great amount of human and time resources our dataset is limited. That is why some of the functional dimensions that naturally occur less frequently are not present in our dataset.

We are training our models to classify texts in 10 functional dimensions[1]: A1 argum, A4 fiction, A7 instruct, A8 news, A9 legal, A11 person, A12 commpuff, A14 research, A16 info, A17 eval (see **Table 2**)[2].

Table 2: Description of training dataset in terms of FTD

FTD	A1	A4	A7	A8	A9	A11	A12	A14	A16	A17
Size	0.14	0.05	0.04	0.25	0.05	0.12	0.17	0.1	0.12	0.09

Also due to the lack of necessary annotation in our dataset, we are not training our models to find texts with partial belonging to the functional dimension. Thus, all of the texts that score positive results are treated equally and marked with the FTD score 1.

[1] The numeration and labels are taken from [Sharoff, 2018].

[2] For complete description of FTDs and how they are defined see [Sharoff, 2018].

3. Experiments

In our research we conducted several experiments involving various features for classification and different hyperparameters of the classification model. The aim of these experiments was to find the most accurate model, which could be used to classify texts of the GICR corpus.

3.1. Classification method

For all models in this study we use the Support Vector Machine (SVM) method for classification. It is a very popular method that is used for various tasks in NLP, such as sentiment analysis [Mullen, Collier, 2004], language recognition [Campbell et al., 2004] and text classification [Sassano, 2003]. SVM also proved to be the best performing method for classification with a dataset similar to ours [Bulygin, Sharoff, 2018].

We conducted our research with SVM in the Python scikit-learn library [Pedregosa, 2011]. During our work we experimented with different kernels of SVM and also with case of letters in texts. The scikit-learn library has 4 build-in SVM kernels: rbf (radial basis function), poly (polynomial), sigmoid, and linear. The linear kernels outperform all the other kernels by a high margin. In this paper we only show the results for the models with linear kernel.

We also looked into the influence of the letter case in training data on the classification performance. The models with the letter case kept as in the original texts generally perform worse than models with letter case converted to lower case, with the only exception being in the functional dimension A12, which contains promotional texts. We attribute this exception to the fact that texts in A12 often contain phrases in upper ('screaming') case, which is a specific feature of this FTD. However, the model without case conversion is unreliable on the complete corpus and in this study we only present results for models with case converted to lower.

3.2. C-value

In the linear SVM model the parameter that is in charge of the strength of regularization is the C parameter. Using low values of C will cause the model to adjust to the majority of data, while using a higher value of C would make model put more attention into the correct classification of each data point [Guido, Muller, 2017]. In our research we build models with different C values. While the default C value is 1, we also looked at models with the C value equal to 10 and 100 (see **Table 3**). We also experimented with a C value less than 1, but the results of these models were much less accurate and they are not present in this paper.

3.3. Feature selection

Feature selection is a process in which a subset of features is selected from all features of training data. The best subset of features contains the least number of features that contribute most to the prediction model [Guyon, Elisseeff, 2003]. Feature selection allows to avoid the overfitting of the model, to reduce training time and

to simplify the model. In our study we implement the basic approach to feature selection. We apply document frequency and only use features that appear at least in 10% of texts of the training data. Models after feature selection have significantly less features than before. For example, a model with character 5-gram features has 235,492 features before selection, and 3,793 features after feature selection.

3.4. Features

In this study we chose character n-grams for features in our models. We made this decision because character n-grams are very useful for text classification [Zhang et al. 2015] and also character n-grams show the best performance in research with training datasets similar to ours [Bulygin, Sharoff, 2018], [Sharoff, 2010].

One of the properties of character n-grams as features is that they can contain not only lexical information about a text, but also morphological, which helps the model to perform better. For our experiment we built models with bigrams, trigrams, 4-grams and 5-grams as features (see **Table 3**). We used scikit-learn preprocessing tools for tokenization and vectorized features using tf-idf technique.

4. Evaluation

For each model we provide 2 metrics: precision and recall. The precision metric tells us how many of the classified documents were classified correctly, while the recall metric shows how many of the texts from the testing data were classified accurately. Only through combination of these metrics one can assess the overall performance of classification.

We named models according to features and to parameters that were set for that model. Thus, model named 'svm-5gr-C10-nfs' should be understood as the model that uses the SVM classification method and character 5-grams as classification features, with C parameter set to 10 and does not use feature selection methods.

Table 3: Evaluation of classification accuracy of models with various parameters

FTD / model	metric	A1	A4	A7	A8	A9	A11	A12	A14	A16	A17
svm-2gr-C1-nfs	precision	0.0	0.0	1.0	0.95	1.0	0.71	0.90	1.00	0.0	1.0
	recall	0.0	0.0	0.50	0.88	0.71	0.31	0.90	0.73	0.0	0.06
svm-3gr-C1-nfs	precision	0.79	0.75	1.0	0.93	1.0	0.71	1.0	0.92	0.75	1.0
	recall	0.50	1.0	0.50	0.86	0.86	0.31	0.88	0.92	**0.35**	0.56
svm-4gr-C1-nfs	precision	0.88	0.75	1.0	0.94	1.0	0.83	0.95	1.0	0.75	1.0
	recall	0.54	1.0	0.50	0.92	0.86	0.31	0.90	0.87	0.16	0.71
svm-5gr-C1-nfs	precision	0.83	1.0	1.0	**0.97**	1.0	0.83	1.0	0.92	0.78	1.0
	recall	0.45	0.67	0.50	0.86	0.86	0.31	0.88	0.92	**0.41**	0.56
svm-3gr-C10-nfs	precision	0.79	0.75	1.0	0.94	0.86	0.38	0.89	0.93	0.53	0.92
	recall	0.58	1.0	0.50	0.92	0.86	0.31	0.85	0.93	0.47	0.65

model \ FTD	metric	A1	A4	A7	A8	A9	A11	A12	A14	A16	A17
svm-5gr-C10-nfs	precision	0.91	0.60	1.0	0.98	0.86	0.46	1.0	1.0	0.80	0.92
	recall	**0.77**	1.00	0.50	0.90	0.86	0.38	0.90	0.93	0.42	0.71
svm-3gr-C100-nfs	precision	0.75	0.75	1.0	0.94	0.86	0.33	0.89	0.93	0.56	0.92
	recall	0.58	1.0	0.50	0.92	0.86	0.31	0.85	0.93	0.47	0.65
svm-5gr-C100-nfs	precision	0.86	0.60	1.0	0.98	0.86	0.46	1.0	1.0	0.80	0.92
	recall	0.73	1.0	0.50	0.90	0.86	0.38	0.90	0.93	0.42	0.71
svm-3gr-C10-fs	precision	0.67	0.75	1.0	0.93	1.0	0.50	0.87	**1.0**	0.53	0.92
	recall	0.55	1.0	**0.67**	0.86	0.86	0.31	0.81	**1.0**	0.47	0.69
svm-5gr-C10-fs	precision	0.68	0.50	1.0	0.95	1.0	0.36	0.88	0.86	0.53	**0.92**
	recall	0.59	1.0	0.67	0.88	0.86	0.25	0.88	**1.0**	**0.53**	0.69

The first four models in Table 3 are basic SVM models with no additional parameters that have different classification features. Out of those four models, the best performing feature appears to be character trigram features and character 5-gram features, where 5-grams slightly outperform trigrams. For the following experiments with SVM parameters we used both these features.

Next, we tested models with different C values. The results show that the increase of C value leads to a better recall score of the model, but lowers the precision. Therefore, the most optimal C value would be 10. Such models are the most balanced, and take into account both precision and recall metrics.

In the end, we implemented feature selection to our best performing models. The reduction of the features helped the model to increase recall score for A7, A14 and A16 functional dimensions. However, these models lost some precision points for some FTDs. In the following experiments with feature extraction and classification of GICR's segments we are going to use 'svm-5gr-C10-fs' model. It is one of our best models and feature selection makes 5-gram interpretation more efficient.

5. Analysis of features

Most of the classifiers used in Machine Learning are a so-called 'black box', because we do not know for sure how the parameters and weights were assigned for the model. However, some of the classifiers, including SVM, are able to show the most valuable features of the model. This can shed some light on how the fitting of the model is performed.

We collected the most valuable features of the 'svm-5gr-C10-fs' model for each of the FTD present in our training corpus. We also provide the most frequent words, where these features appear (see Table 4).

Table 4: The most significant classification features for each FTD

FTD	Features	Words
A1	' соци', 'оказа', ' прич', 'ителя', ' наро', ' нам ', ' бога', ' нет ', ' чем ', ' они '	социальной, оказания, причем, представителя, международного, богатства, показатели, доказательства, оказались, причин, заместителя, народа
A4	' ка , ', ' прос', ' , - ', ' , и ', ' а , и", ' его ', 'казал', ' глаз', ' , не", '. он '	человека, просто, его, сказал, глаза, века, казалось, показал, некоторые
A7	'нстру', ' доба', 'добав', 'форма', ' вас ', ' жела', ' поро', 'если ', ' если', ' запр'	конструкции, добавить, год, информации, желание, пород, если, запрос, инструментов, порой, запрещено
A8	'ября ', ' новы', 'сказа', 'сообщ', ' моск', ' сооб', 'явил ', 'заяви', 'аявил', ' заяв'	сентября, новых, сказал, московских, сообщения, заявил, октября, основы, появились
A9	'ветст', 'должн', 'етств', ' стат', ' зака', 'мать ', 'стать', ' рабо', ' или ', 'федер'	должны, соответствии, статьи, заказ, принимать, работы, ответственности, должностных, федерального, статус
A11	'много', ' лет ', ' свое', ' нас ', 'перед', 'лись ', ' . к ', 'стно ', ' меня', ' мне '	оказались, появились, остались, находились, известно, совместно
A12	'знако', ' азмер ', ' крас', 'овый ', ' для ', ' прод', 'наком', ' высо', ' сайт', 'ство '	признаков, красоты, красный, красивый, новый, знаком, продукции, высокой, сайте, количество, большинство
A14	' она ', ' зада', 'ений ', 'больш', ') . ', ' язык', ' расс', 'ости ', ' боль', ' . и '	задачи, языка, рассмотрения, больше, отношений, решений, изменений, расследования, деятельности
A16	' зако', 'начал', 'изма ', 'закон', ' . а ', 'нный ', ' . в ', ' прин', 'века ', ' жела'	закона, начала, механизма, данный, принять, человека, желание, организма, современный единственный
A17	' игра', ' хотя', 'хотя ', 'смотр', 'разу ', 'овски', ' мало', ' без ', 'стати', ' книг'	играть, рассмотрения, сразу, московский, кстати, книги, разумеется, банковские

As shown in Table 4, features that are the most significant for the FTD are appearing in words that are often associated with texts of this functional dimension. For example, 'заявил' in A8 and 'должностных' in A9.

6. Applying FTD classifier to the GICR corpus

The GICR corpus contains over 2 million documents with over 20 billion words. The corpus is split into several segments, based on the source of the document. We chose the 'svm-5gr-C10-fs' for the GICR classification, since 5-grams and the C value of 10 turned out to be the best overall performing classification options. We also chose to filter features, because it speeds up the classification and models with a reduced number of features are usually more reliable.

Before we applied our 'svm-5gr-C10-fs' model to the segments of the GICR corpus, we decided to test it on the raw dataset of the GICR texts from the 'livejournal' segment, since our model was only tested on our training corpus. For this experiment, we randomly picked 100 texts from 'livejournal' subcorpus and annotated them manually. Then we evaluated the 'svm-5gr-C10-fs' model on these texts. The overall

precision averaged around 75% and overall recall was 51%. For the two most represented functional dimensions in 'livejournal' segment A1 and A11 these metrics scored 75% precision, 47% recall for A1 and 83% precision, 62% recall for A11. We considered the performance of this model adequate, so we applied it for the classification of the GICR corpus (see Table 5).

Table 5: Classification of GICR segments in terms of FTD

FTD / segment	A1	A4	A7	A8	A9	A11	A12	A14	A16	A17
Livejournal.com	0.39	0.02	0.003	0.09	0.002	0.42	0.01	0.001	0.02	0.05
Blogs.mail.ru	0.52	0.02	0.01	0.004	0.0004	0.39	0.01	0.0003	0.006	0.04
magazines.russ.ru	0.16	0.33	0.0	0.003	0.01	0.24	0.0	0.06	0.14	0.05
News	0.04	0.0001	0.0002	0.92	0.003	0.003	0.002	0.002	0.02	0.004
Vk.com	0.71	0.04	0.003	0.01	0.001	0.19	0.03	0.0003	0.01	0.007
Total GICR	0.45	0.02	0.007	0.11	0.002	0.30	0.03	0.005	0.02	0.05

The classifier was able to mark most of the texts in each segment, though some of the texts were left unlabeled. These texts are not taken into account in Table 5.

In the 'livejournal' segment we see the dominance of A1 (argumentative blogs) and A11(personal stories) functional dimensions. Both of these FTDs are common for social networks, and it is not a surprise that they compose most of the 'livejournal' segment of GICR.

The 'blogs.mail.ru' segment is quite similar to the 'livejournal' segment, as it also has A1 and A11 FTDs comprising it. However, it is expectable, since both sites are platforms for blogs and, hence they have similar type of texts.

The 'magazines.russ' segment consists of various journals with different style of articles. This can be seen in our results. This segment is the most well-rounded, with no dimension being severely dominant.

The 'news' segment is composed of articles from ria.ru, lenta.ru and rosbalt.ru. The most represented dimension here is A8. A portion of the texts in this segment is from the A1 (argumentative blogs) FTD, which is common for news texts.

The 'vk' segment is a social network segment. It is also dominated by the A1 and A11 functional dimension.

An interesting question that comes up during the FTD research is how functional dimensions correlate with linguistic features. We used a script that extracts linguistic features on the A1 and A8 subsets. The script was adapted for Russian by Serge Sharoff from MultiDimensional Analysis [Biber, 1988].[3] The subsets were classified by our model from the 'livejournal' segment of GICR. In our experiment we looked at two features: the verbs in the present tense and in the past tense. The results show that verbs in the present tense are much more common in the A1 dimension than in the A8 dimension, with median values 0.03490829 and 0.02912898 respectively.

[3] https://github.com/ssharoff/biberpy

However, the verbs in the past tense are more frequent in the A8 dimension with median value 0.03572108 and much less frequent for the A1 dimension with median value 0.01967835. This opens the possibility to compare the use of language in argumentative opinion pieces vs reporting news. More research is still required.

7. Conclusion

In this paper we presented an experiment during which we tested several classification features and parameters to find the optimal classification options for the GICR dataset. The resulting model uses character 5-gram features, has C-value of 10 and uses the feature selection technique, where features are filtered by document frequency. This model was used for the annotation of the segments of the GICR corpus. Furthermore, we looked at the most significant features of our model for each FTD and compared these 5-grams t0 the most frequent words of the training corpus, in which these features can be found.

In further studies we would like to continue our experiments with GICR annotation. One of the possible lines of research is the correlation between text-internal linguistic features and text-external genre classification. The original idea comes from Douglas Biber's Multi-Dimensional analysis [Biber, 1986]. The MD analysis was also implemented for the English web texts in [Biber, Egbert, 2016]. Similar research was conducted for Russian in [Katinskaya, Sharoff, 2015], where the researchers used FTD classification and compared it to the MD analysis. This study showed very promising results and we would like to apply this knowledge to the GICR corpus.

Another interesting research area concerns related text classification tasks. We have not experienced considerable issues with detecting spam, most of it was classified as A12, Promotion. However, it'd be very interesting to investigate deviations from the prototypes (such as a newspaper report) as caused by spam in the social networks.

References

1. *Adamzik K.* (1995), Textsorten—Texttypologie, Eine kommentierte Bibliographie, Nodus, Münster.
2. *Biber D.* (1986) Spoken and written textual dimensions in English: resolving the contradictory findings. Language, Vol. 62, pp. 384–414.
3. *Biber D., Egbert J.* (2016) Register Variation on the Searchable Web: A Multi-Dimensional Analysis. Journal of English Linguistics, 44(2), pp. 95–137.
4. *Bulygin M., Sharoff S.* (2018) Using Machine Translation for Automatic Genre Classification in Arabic. In Proc Dialogue, Russian International Conference on Computational Linguistics.
5. *Campbell W., Singer E., Torres-Carrasquillo P., Reynolds D.* (2004) Language recognition with support vector machines. In Proc. ODYS, pp. 41–44.
6. *Görlach M.* (2004), Text types and the history of English, Walter de Gruyter.
7. *Guido S., Muller C.* (2017), Introduction to Machine Learning with Python, O'Reilly Media, Inc, pp. 57–58.

8. *Guyon I., Elisseeff A.* (2003), An introduction to variable and feature selection, J. Mach. Learn, pp. 1157–1182.
9. *Katinskaya A., Sharoff S.* (2015), Applying Multi-Dimensional Analysis to a Russian Webcorpus: Searching for Evidence of Genres. In Proc BSNLP, Sofia.
10. *Mullen T., Collier N.* (2004) Sentiment analysis using support vector machines with diverse information sources, in Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 412–418.
11. *Piperski A., Belikov V., Kopylov N., Selegey V., and Sharoff S.* (2013), Big and diverse is beautiful: A large corpus of Russian to study linguistic variation, In Proc 8th Web as Corpus Workshop (WAC-8).
12. *Santini M., Mehler A., Sharoff S.* (2010), Riding the rough waves of genre on the web, Genres on the Web: Computational Models and Empirical Studies, Springer, Berlin/New York.
13. *Sassano M.* (2003), Virtual examples for text classification with support vector machines, In Proceedings of Empirical Methods in Natural Language Processing, pp. 208–215.
14. *Sharoff S., Wu Z., Markert K.* (2010) The Web Library of Babel: evaluating genre collections. In Proc Seventh Language Resources and Evaluation Conference, LREC, Malta.
15. *Sharoff S.* (2018), Functional text dimensions for annotation of web corpora, Corpora.
16. *Zhang X., Zhao J., LeCun Y.* (2015), Character-level convolutional networks for text classification, In Advances in Neural Information Processing Systems, pp. 649–657.