

Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2020”

Moscow, June 17–20, 2020

COMBINING FACTS, SEMANTIC ROLES AND SENTIMENT LEXICON IN A GENERATIVE MODEL FOR OPINION MINING

Feldman D. G. (daniil.feldman@phystech.edu)

Moscow Institute of Physics and Technology, Moscow, Russia

Sadekova T. R. (str12.01.94@gmail.com)

Lomonosov Moscow State University, Moscow, Russia

Vorontsov K. V. (vokov@forecsys.ru)

Moscow Institute of Physics and Technology, Moscow, Russia

Opinion mining is a popular task, that is applied, for example, to determine news polarisation and identify product review classes. Our task is unsupervised clusterization of opinionated texts, in particular news on political events. Many papers that tackle this issue use generative models based on lexical features. Our goal is to determine the entities defying an opinion amongst lexical, syntactic and semantic features as well as their compositions. More specifically, we test the hypothesis that an opinion is determined by the composition of the mentioned facts (SPO triples), the semantic roles of the words and the sentiment lexicon used in it. In this paper we formalise this task and prove that using a composition of the above features provides the best quality when clusterising opinionated texts. To test this hypothesis we have gathered and labelled two corpuses of news on political events and proposed a set of unsupervised algorithms for extracting the features.

Keywords: opinion mining, SPO triplets, Fillmore roles, sentiment lexicon

DOI: 10.28995/2075-7182-2020-19-283-298

КОМБИНИРОВАНИЕ ФАКТОВ, СЕМАНТИЧЕСКИХ РОЛЕЙ И ТОНАЛЬНЫХ СЛОВ В ГЕНЕРАТИВНОЙ МОДЕЛИ ДЛЯ ПОИСКА МНЕНИЙ

Фельдман Д. Г. (daniil.feldman@phystech.edu)

Московский физико-технический институт, Москва, Россия

Садекова Т. Р. (str12.01.94@gmail.com)

МГУ имени М. В. Ломоносова, Москва, Россия

Воронцов К. В. (vokov@forecsys.ru)

Московский физико-технический институт, Москва, Россия

1. Introduction

Every important political event is vastly covered in the news. Most sources provide polarised texts expressing the opinion of one of the sides. As a result, readers only get to know one side of the problem. We would like to provide them with a tool to clusterize news feed according to the opinions they express. Often there are more than two opinions on an issue, and a more general approach suggests finding the number of opinions as well. We, however, will be solving a simpler problem where this variable is given.

In this paper we study different approaches to classify opinionated texts on a given news story. We work with datasets of news covering political events. The question we answer is what computationally seizable entities define the opinion of a text. In contrast to the work done traditionally in opinion mining, which focuses on proposing a more complex generative model, in this work, we focus on determining the best features for classification and finding the entities that authors use to express opinions.

Most studies use lexical features (words or sentiment lexicon) as tokens and extract dependencies on a word-level. We assume that syntactic and semantic patterns combined capture the essence of the author's opinion. To be more specific, we propose three entities: subject-predicate-object (SPO) triples, semantic roles and sentiment lexicon. To understand how they capture opinions that are indistinguishable on a word-level, let us consider two extracts from actual news covering the enterprise nationalisation in LPR and DPR (Lugansk and Donetsk People's Republics). These texts have similar word distributions but the mentioned semantic and syntactic features are used differently.

...Президент Петр Порошенко заявил, что Россия де-факто конфисковала украинские предприятия, которые находятся на неподконтрольной Киеву территории. Сегодня ДНР и ЛНР «национализировали» украинские предприятия ... При этом Кремль защитил конфискацию предприятий в ЛДНР ... Украина требует расширить санкции ... За все эти действия обязательно наступит наказание. Украина потребует расширения санкций на тех, кто украл украинские предприятия ...

...По словам Захарченко, Киев встретит свой «ужасный конец»... Киев возьмется за ум, и в целях спасения собственной промышленности снимет блокаду... Обстановка, которую искусственно создала Украина с блокадой Донбасса, вынудила ... кошмарит свой народ ... если в Киеве были приняты какое-либо постановление ... положительные результаты, как для республик, так и для России ... Если им удастся сместить Порошенко и при этом не развалить Украину, то все вернется на свои места...

The words *Poroshenko*, *Russia*, *Ukraine* and others are used in both texts, so we wouldn't be able to tell the opinions apart based on lexical features. One way to solve this is to use subject-predicate-object triples. For example, in the first text *Poroshenko* is used as a subject, but in the second one as an object. The second approach we mentioned is using semantic roles. For example, the word *Russia* is an agent in the first text and a patient in the second one. Lastly, we use sentiment lexicon to tell the opinions apart: the first text uses such negative lexicon as *occupation*, *steal* (aggressor) while the second one has *hostage*, *threat* (victim). To clarify, word distributions do matter but they differ mostly in sentiment lexicon. In this paper we study what feature: SPO triples, semantic roles or sentiment lexicon captures the differences between opinions more accurately. In [section 3](#) we describe these features and the unsupervised algorithms we use to extract them in detail. In [section 4.1](#) we describe the probabilistic models we use for opinion clusterization. In [section 5](#) we present the openly available datasets of news labelled with opinions we have gathered. [Section 5.3](#) contains comparisons of models built on SPO triples, semantic roles, sentiment lexicon and their combinations. Finally, we present our conclusion that an opinion, in fact, is best defined as a combination of all three features.

2. Related work

Opinion mining has been vastly studied in recent years. A general survey of methods is presented in [\[2\]](#) and [\[3\]](#). Earlier works [\[1\]](#), [\[4\]](#) focused on mining opinions in product reviews, but in more recent papers focus has shifted towards texts on political events and the issue of political polarisation [\[5\]](#), [\[6\]](#), [\[12\]](#). Most of the discussed works rely on probabilistic models, which are described in [\[2\]](#). Topic models were used both in supervised [\[5\]](#) and unsupervised approaches [\[6\]](#). When analysing news feed, supervised approaches are hard to apply as big datasets with labelled texts are difficult to come by and cannot be gathered while the news are still relevant, so a supervised approach was not an option for us. All in all, unsupervised generative models are the most preferred and popular in opinion mining [\[5\]](#), [\[4\]](#), [\[14\]](#). Some works tackle a more

general problem of finding topics and opinions simultaneously. The authors of [7] propose a topic-aspect model that mines topics and aspects, where the latter can be interpreted as opinions. We will be focusing on a more localised problem of clustering opinionated texts on a given political event or topic. In other words, we will be building the second layer of a hierarchical topic model of the news feed, which corresponds to opinions. Unsurprisingly, the overwhelming majority of opinion mining studies worked with English texts, while we will be experimenting with Russian news.

SPO triples (or facts) have been used for opinion mining, and they allowed to increase quality in similar problems. For example, the study [8] was aimed at solving a different task: ontology mining. Its authors built a similar generative model (LDA). They proved that using SPO triples to build hierarchical topic models provides a gain in quality. That gave us the idea that a similar heuristic could be useful in opinion mining.

Sentiment analysis is close to opinion mining. Sentiment lexicon is a strong indicator of an opinion and can be accurately extracted without a train dataset, making it a popular tool for opinion mining. Many studies use polarised words to classify product reviews and political texts. The most basic approach is lexicon-based, but its main drawback is that context can massively affect word polarity. A necessary enhancement is rule-based approaches [16] that take into account neighbouring words, their part of speech, sentence role and other features. Works such as [17] propose a system based on a large set of heuristical rules, and our model follows a similar approach.

Semantic roles are powerful in capturing connections in sentences and they have been used for solving many NLP tasks such as question answering, information extraction and information search. Study [15] showed that they can provide a significant quality increase in opinion mining. There are several approaches for extracting semantic roles, most of them use some type of neural network architecture [19], [20], [21] and a hand-accessed database of semantic frames, such as FrameNet or VerbNet [20], [21], for training. A model for extracting semantic roles in Russian was proposed by Shelmanov and Devyatkin in [18]. It used a similar neural networks approach and the Russian database FrameBank, similar to FrameNet.

SPO triples and semantic roles have been used for opinion mining. However, a combination of semantic and syntactic features have not.

To build topic models we will be following an approach called additive topic modelling regularizers (ARTM), a technique that allows to configure different topic models by adding regularizers. It is described in detail and analysed in [9].

3. Features

To build the composite probabilistic model we will be using several syntactic and semantic features. This section will provide their description and describe the unsupervised algorithms we proposed to extract them.

3.1. SPO triples

In subject-predicate-object triples predicates are the words that express situations, they are usually verbs or verbal nouns. Naturally, in order to extract SPO triples,

we would first have to find predicates and then find subjects and objects connected to them. In order to do that we will be building syntactic dependency

trees, an example for the sentence «Радикалы зажгли файеры возле украинской дочки Сбербанка» is given in **Figure 1**. At its nodes are words connected with edges representing syntactic tags in the Universal dependencies format.

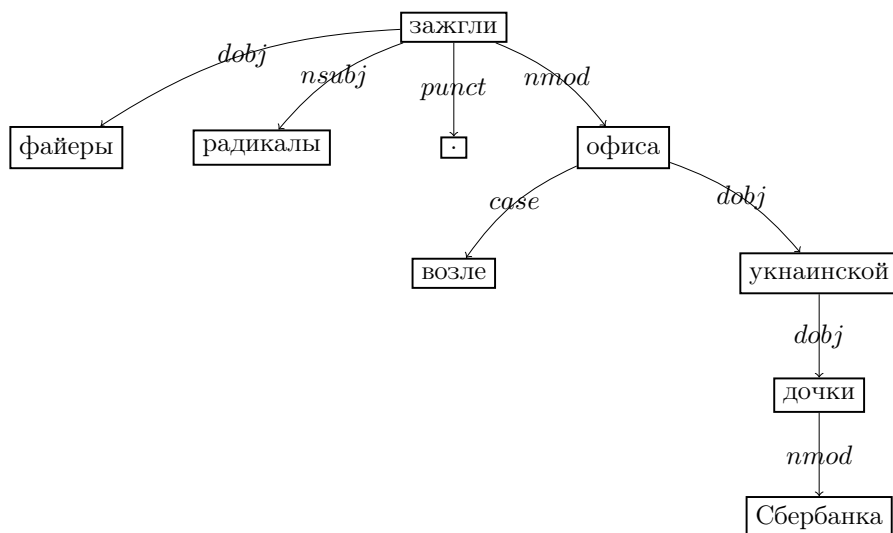


Figure 1: Dependency tree example

To extract the triplets we take into account a number of features such as part of speech, position, syntax tag and others. The most basic SPO triplets are noun-predicate-noun. Besides explicit triplets there are some implicit examples that do not contain a verb or a verb noun, such as noun-noun triplets.

We developed a set of heuristics allowing us to capture the following types of triplets:

- Explicit triplets: noun-verb-noun.
Example: *the congress passed a law* → (congress, pass, law)
- Noun-noun triplets.
Example: *president Putin* → (Putin, is, president)
- Adverb triplets.
Example: *The meeting held by Navalny...* → (Navalny, hold, meeting)
- Adjectives triplets.
Example: *Ukrainian branch* → (branch, is, Ukrainian)

To build the dependency trees we used a dockerized version of Google's SyntaxNet parser¹ pre-trained on a corpus of Russian texts.

¹ <https://github.com/tensorflow/models/tree/master/syntaxnet>

3.2. Fillmore roles

Semantic roles capture the meaning of words and show what the author meant to express. They are a way to describe numerous word relations by a limited set of states. There are many different sets of semantic roles of various sizes. On one end of the spectrum are domain-specific roles such as `From_Airport`, `To_Airport`, `Depart_Date`, they are fixed to a frame (`Flight`) and often to a small set of words. They are not suitable as features as their distributions in any given text are very sparse. On the other end of the spectrum are macro-roles: `Proto_Agent` and `Proto_Patient`, they are too general and usually match subjects and objects. In between lay many sets of semantic roles such as Fillmore's nine: `Agent`, `Experiencer`, `Instrument`, `Goal`, `Location`, `Object`, `Source`, `Time`, and `Path`. Their description is given in [Table 1](#). We decided to choose Fillmore's [13] set of roles as they are cross-domain and describe complex semantic relations.

To extract Fillmore roles we will be using the model proposed by Shelmanov A. O. and Devyatkin D. A. in 2018 [18]. It uses a neural network using syntactic features gained from a dependency tree (built with `SyntaxNet`). The model extracts multiple semantic roles which we then clusterise into Fillmore's 9 roles.

Table 1: Fillmore's semantic roles

Role	Description
Agent	The instigator of the action denoted by the predicate
Patient	The 'undergoer' of the action or event denoted by the predicate
Theme	The entity that is moved by the action or event denoted by the predicate
Experiencer	The living entity that experiences the action or event denoted by the predicate
Goal	The location or entity in the direction of which something moves
Benefactive	The entity that benefits from the action or event denoted by the predicate
Source	The location or entity from which something moves
Instrument	The medium by which the action or event is carried out
Locative	The specification of the place where the action or event is situated

3.3. Sentiment lexicon

Opinions on an object is often supported by sentiment lexicon, and under sentiment we will be understanding two polarities: positive and negative. The problem of extracting sentiment lexicon in news can be broken down into two tasks: gathering a dictionary with sentiment lexicon and tagging words with respect to their context in the texts with +1, -1 or 0.

As a basis dictionary we used the one collected by the resource Linis Crowd ([10]), which was gained through crowd-accessing of texts on political and social topics. The dictionary contains 2,454 words tagged with sentiment. We then enriched the dictionary using an approach that suggests that synonyms and hyponyms of a sentiment

word have the same polarity while antonyms have the opposite. For that we used Ru-WordNet² [11] and widened the dictionary to 3,419 words.

When tagging words with sentiments we took their context into account. There are several ways to do that but the best one is syntactic dependency trees. We used the following rules to tag words:

- If a tagged word is a noun, an adjective or an adverb, its parents are tagged with the same polarity.
- If a verb is tagged, subjects and objects connected to it are tagged as well.
- If a negative particle is a child of a tagged word, its polarity is changed to the opposite.

4. Multimodal opinion mining model

Our algorithm consists of several steps. Firstly, we build syntactic dependency trees of each sentence in the document corpus using SyntaxNet. The same trees are then used in all three algorithms that extract SPO triples, Fillmore roles and sentiment lexicon. The features are then broken down into 5 modalities, section 4.2 covers the process. Those modalities are used to built a regularised topic model. Finally, we train the topic model on a corpus of documents and split them into clusters corresponding opinions. Figure 2 shows a general plan of the algorithm. Next we will be describing every step in detail.

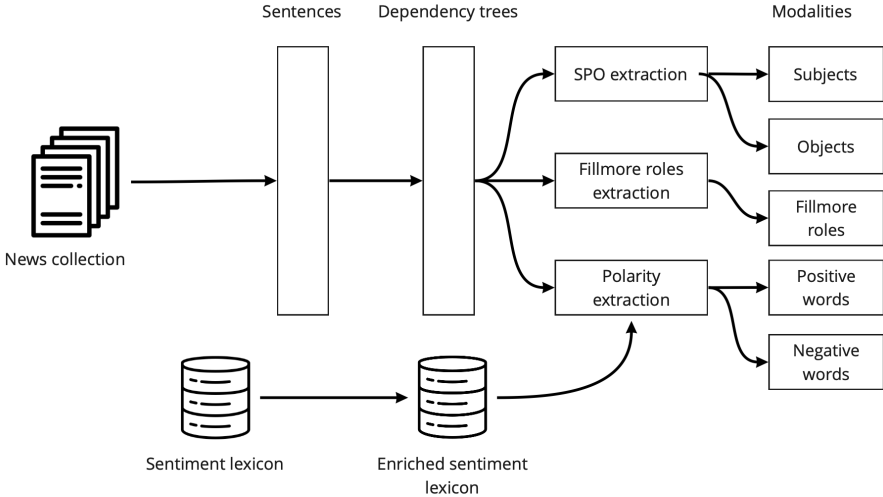


Figure 2: Modalities extraction algorithm

² <http://www.ruwordnet.ru/ru>

4.1. ARTM topic model

In the current section we will describe the basics of additive regularization of topic models. Let D be the corpus of documents and W the set of words (tokens) in them. We will consider every document a bag of words. We assume that each word relates to some topic from T and the corpus is an i.i.d. $(w_i; d_i; t_i)_{i=1}^n$ from a distribution $p(w; d; t) \in W \times D \times T$. Having also assumed that the appearance of a word w in a document depends only on the topic, we can draw:

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d) = \sum_{t \in T} \varphi_{wt}\theta_{td} = \Phi \cdot \Theta$$

Φ and Θ are stochastic topic-word and document-topic matrices, our goal is to find them. To do that we will be maximizing the likelihood logarithm with probability distribution constraints:

$$L(\Phi, \Theta) + R(\Phi, \Theta) = \ln \prod_{d \in D} \prod_{w \in d} p(w|d)^{n_{dw}} + R(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \sum_{t \in T} \varphi_{wt}\theta_{td} + R(\Phi, \Theta)$$

Instead of words we can break the documents down into of several types of tokens called modalities (i.e. subjects, objects, polarised words). In this case the equation takes the form:

$$L(\Phi, \Theta) + R(\Phi, \Theta) = \sum_{m \in M} \tau_m \sum_{d \in D} \sum_{w \in d} n_{dw} \sum_{t \in T} \varphi_{wt}\theta_{td} + R(\Phi, \Theta)$$

Here M is the list of all modalities. W^m is the dictionary of m-th modality. By adjusting weights τ_m we control the influence of each modality. The resulting optimisation task is:

$$\begin{aligned} \min_{\Phi, \Theta} & \sum_{m \in M} \tau_m \sum_{d \in D} \sum_{w \in W^m} n_{dw} \sum_{t \in T} \varphi_{wt}\theta_{td} + R(\Phi, \Theta) \\ \text{s.t.} & \sum_{w \in W^m} \varphi_{wt} = 1, m \in M; \varphi_{wt} \geq 0 \\ & \sum_{t \in T} \theta_{td} = 1, m \in M; \theta_{td} \geq 0 \end{aligned}$$

Here $R(\Phi, \Theta)$ is the regulariser we add to the likelihood. If $R(\Phi, \Theta) = 0$ it turns into a PLSA model. In this case the problem has an infinite number of solutions, so regularisers provide additional constraints. $R(\Phi, \Theta)$ is usually a sum of regularisers with coefficients: $\sum \tau_i R_i(\Phi, \Theta)$. We will be using the following regularizers in our work:

- Smooth/sparse regularizer:

We presume that every document and every word relates to a small number of topics, so the distributions $\varphi_t = (\varphi_{wt})_{w \in W}$; $\theta_d = (\theta_{td})_{t \in T}$ should be sparse. At the same time there are some common vocabulary topics that are present in every document, their distributions are smooth. Common vocabulary does not carry much information for our experiments, so we add a smooth regulariser for

those words to be gathered in common vocabulary topics. Other topics will then have sparse distributions where only a few words are assigned high probabilities. Those words can be called kernel words. The model we build has one extra topic that holds common vocabulary terms.

- Decorrelation regulariser:
In opinion mining it is necessary that the opinions found in the text corpus differ significantly, otherwise we cannot classify the texts. We add the decorrelation regulariser so that is sparse.

4.2. Modalities

We will be building probabilistic models based on five modalities: $M = \{M_s, M_o, M_r, M_p, M_n\}$. The first two modalities are subjects and objects, derived from SPO triples. At first we used the whole triples as a modality, but the distribution of terms over texts in any given corpus was too sparse: most triples were unique to the collection. Usage of SPO triples for opinion mining was based on the heuristic that news with similar lexicon had differing distributions of subjects and objects, that is why we decided to use them as the two modalities. Let $T_d = \{(s_1, p_1, o_1), \dots, (s_n, p_n, o_n)\}$ be the list of all triplets in document d , then we create two lists $\{s_1, \dots, s_{n_s}\} \in W^s$ and $\{o_1, \dots, o_{n_o}\} \in W^o$ and define n_{dw} as the number of triples containing w :

$$n_{dw_s} = \sum_{(s,p,o) \in T_d} [s = w], w \in W^s$$

$$n_{dw_o} = \sum_{(s,p,o) \in T_d} [o = w], w \in W^o$$

The next modality M_r corresponds to roles. Its dictionary W^r contains all pairs (w, r) , $w \in W$, $r \in R$, where W is the corpus vocabulary and R is the set of nine Fillmore roles. Here n_{dw} is calculated as the number of times the pair (w, r) occurs in document d . The last two modalities M_p and M_n correspond to sentiment lexicon. W^p is a dictionary that contains all positively polarised words while W^n contains all negatively polarised words.

4.3. Assigning opinions

Having obtained the matrix θ with topic distributions for every document we have to assign some opinion to every text, that is to clusterise them. In order to do that we will consider θ_d as a vector of features for d . We can say that feature x_i shows “how much” of topic t is contained in d . If we assume that every text expresses a single opinion, it can be defined as

$$o_d = \arg \max_t \theta_d$$

It must be noted that an opinion does not necessarily express a positive or negative attitude to an entity, as there can be any number of opinions in a text corpora. It is rather a combination attitudes on different entities.

5. Experiments

To evaluate our algorithm we have collected and labelled two corpuses of news:

1. 82 news considering enterprise nationalisation in LPR and DPR. The texts average at 200 words. They were extracted from multiple news sources: Russian as well as Ukrainian. We selected texts expressing two opinions: Moscow’s opinion and Kiev’s opinion, most texts on the topic belong to one of them.
2. 220 news considering Donald Trump’s decision of quitting the Paris Climate Agreement. The text’s sizes once again averaged at around 200 words. The news were equally distributed between two opinions: one of Trump’s supporters, those who oppose him (such as Elon Musk).

The corpuses were labelled by two independent accessors who were given selected news on a topic. They first read the whole collection and determined the number of opinions present. Then every text was marked with an opinion. The two accessors reached an agreement with an average rate of 91% when assigning opinions. An additional accessor was brought to evaluate uncertain texts. When collected, the corpuses were being extended until news began to be derivative from those already added. The datasets are available for public use at [our repository](#)³. From here on we will be referring to the datasets as Corpus 1 and Corpus 2.

To evaluate clusterisation quality over two classes we will be measuring precision, recall and F1-score.

5.1. Adjusting hyperparameters

Our model has several hyperparameters:

- Weights of modalities $\{\tau_m\}$, $m \in \{s, o, r, p, n\}$, $\sum_m \tau_m = 1$
This is the most important parameter—the weight distribution of modalities that defines an opinion. One of the goals of our research is to get the optimal distribution of features: SPO triples, semantic roles and sentiment lexicon that provides the best clusterisation quality and thus defines an opinion.
- Regularization coefficients τ
These coefficients determine how sparse or smooth the resulting distributions in Φ and θ are.
- Minimal TF for modalities
It is possible to take into account words with term frequency above a threshold.

To find the optimal hyperparameters we follow the steps:

1. Fix the regularization parameter at $\tau = 1.0$ and adjust the minimal TF for all modalities. We start with this hyperparameter as it takes only a small discrete set of values.

³ https://github.com/newfteddy/opinion_mining_features/tree/master/data

2. With fixed minimal TF optimise the regularization coefficients for each modality. τ has a noticeable effect on overall performance.
3. Find the optimal modalities weights $\tau_m, m \in \{s, o, r, p, n\}$ with all other parameters fixed.

The first step yielded the optimal minimal TF for filtering the dictionaries M_i : for semantic roles and SPO triplets it is 2, for sentiment lexicon—1. Results of the second step: optimisation with regard to τ are presented in **Figure 3** for the first corpus and in **Figure 4** for the second corpus.

Each plot shows the value of F1-score of a model trained on a single feature with regard to the parameter τ . The results may seem different at first, but they are in fact quite similar. On both corpuses there is a clear maximum point around 0.8 for all three features. The difference is that on the second corpus there is a distinct quality prevalence of SPO and Fillmore roles features over the sentiment lexicon model. One way to explain this is that the second dataset is significantly larger then the first one so there is better convergence. In any case, the goal of our research is not to find out what feature works best to classify opinionated texts but wether their combination provides a noticeable increase of quality.

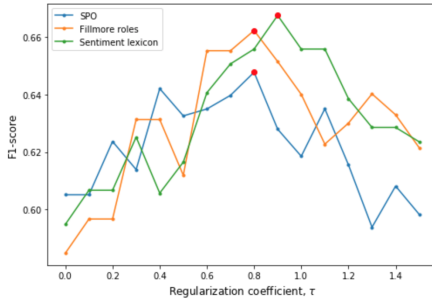


Figure 3: Optimal τ for Corpus 1

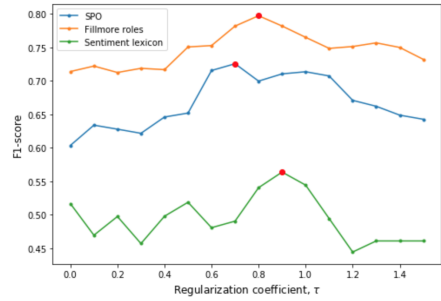


Figure 4: Optimal τ for Corpus 2

5.2. Lexical baseline

To demonstrate the sensibility of using probabilistic models with complex features we will compare our model with a simple bag-of-words baseline. First, we represent each document with a vector of its words.

Thus the corpus can be viewed as a matrix, where each column is a feature vector $\{tf\text{-idf}(w, d)\}_{W \times D}$. Next, we clusterize the documents using the k-means algorithm.

The result of the k-means depends on the initial point, so to obtain a more representative result we conducted 100 experiments with the baseline and averaged the results for each corpora. **Figure 5** shows the average f1 measure for the baseline algorithm. The number of iterations show how many experiments were conducted. The average quality is 0.67 on Corpus 1 and 0.72 on Corpus 2. The result variance is around 0.5, it is higher on Corpus 1 due to its smaller size.

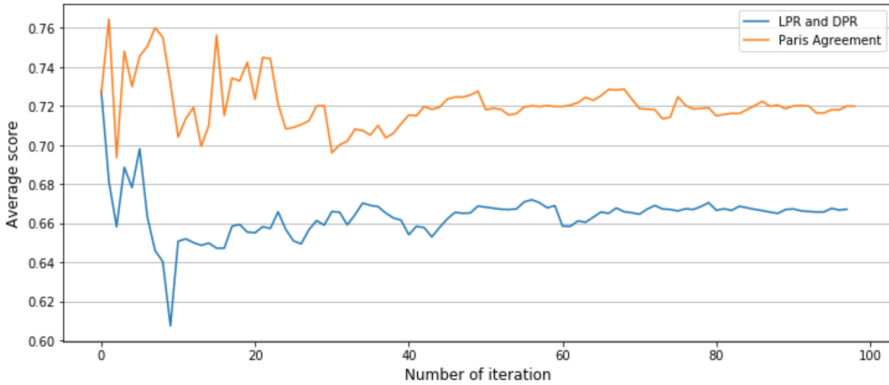


Figure 5: Average F1-measure for lexical baseline

5.3. Evaluation results

To evaluate clustering we will be measuring precision, recall and F1-score. Both datasets have two classes, so the quality for a constant model would be 0.5. The goal is to determine what defines an opinion. In order to do that we will be training our probabilistic model with different sets of modalities. They will be comprised of:

- Subject-predicate-object triplets (SPO)
- Fillmore roles (FR)
- Sentiment lexicon (Sent)

In the beginning we train probabilistic models with a single feature to see how well they can clusterise opinionated news on their own. Then we try combining features in pairs: SPO triples together with Fillmore roles (SPO+FR), SPO triples with sentiment lexicon (SPO+Sent) and Fillmore roles with sentiment lexicon (FR+Sent). Finally, we evaluate the model using a combination of all three features. The results are presented in **Tables 2** and **3**.

Table 2: LPR and DPR enterprises

Modalities	Pr	Rec	F1
TF-IDF	0.51	0.95	0.67
SPO	0.59	0.7	0.64
FR	0.86	0.49	0.65
Sent	0.69	0.57	0.66
SPO+FR	0.86	0.68	0.76
SPO+Sent	0.83	0.78	0.81
FR+Sent	0.9	0.52	0.67
SPO+FR+Sent	0.77	0.97	0.86

Table 3: Trump leaving the Paris Agreement

Modalities	Pr	Rec	F1
TF-IDF	0.57	0.97	0.72
SPO	0.56	0.99	0.72
FR	0.67	0.97	0.79
Sent	0.56	0.55	0.55
SPO+FR	0.72	0.99	0.83
SPO+Sent	0.57	0.99	0.72
FR+Sent	0.73	0.97	0.83
SPO+FR+Sent	0.77	0.94	0.85

The resulting models that used a combination of all three features significantly outperformed the lexical baseline. However, single-feature models showed similarly low quality.

On the first corpus all three models trained with a single feature showed quality around 0.65. This shows that in the first dataset a single syntactic or semantic feature is unable to separate news relating to different opinions, which is not surprising. However, the features gave false classifications on separate groups of news. This fact allowed models trained on a combination of two features to show a significant growth of quality with a combination of SPO triples and sentiment lexicon showing the F1-score of 0.81. Combining all three features yields the best result extending the quality growth and bringing the F1-score up to 0.86 mainly on account of notably increasing recall.

Analysing performance on the second corpus, we can see a similar pattern. Of the three single feature models only the one using Fillmore roles showed decent quality. The SPO model had a high recall score but very low precision, while sentiment lexicon had poor performance overall. Once again, combining the features increased the F1-score in all three cases. We note very interpretable results in the sense that combining the two best features yielded the best results of all pairs raising the quality to 0.83. Similar to Corpus 1 the best score of 0.85 was reached by combining all three features bringing precision up significantly from single-feature models.

The resulting scores were reached with one point of the modalities weights τ_m distribution. A logical question arises: how stable are the optimal points of this hyperparameter? To answer it we paired the features and conducted experiments with varying weights distribution. For example, for the feature pair SPO triples and Fillmore roles we set the weight of the latter with values $\tau_r \in \{0, 0.05, 0.1, \dots, 1\}$ and the weight of the triples as $1 - \tau_r$. A similar procedure was done with other pairs of features. Results are illustrated in [Figure 6](#) for Corpus 1 and [Figure 7](#) for Corpus 2. Firstly, let us examine [Figure 7](#). For the pair roles-SPO (third line) the left end shows a model trained solely with semantic roles, which has the F1-score of 0.79, and the right end reflects a model using only SPO triples with the F1-score of 0.72. As we move right, adding more weight to the SPO modalities, quality starts increasing to the point of weight distribution of $\{\tau_r = 0.7, \tau_s + \tau_o = 0.3\}$ and gradually decreases from there. A similar result is observed with other feature pairs, allowing us to conclude that the optimum is stable with regard to feature weights. Moving on to [Figure 6](#) we see, that on the first corpus the optimal solution is not as stable. All in all, a similar pattern appears: moving from right to left increases quality to a point, just not as gradually. Tweaking the weights by 0.1 can change the score notable. There are several ways to explain this behaviour. Firstly, the dataset itself is quite small so results are less stable. Secondly, all the single-feature models have relatively low performance on their own. Further investigation of this issue requires widening Corpus 1 and gathering additional datasets which we leave for future work.

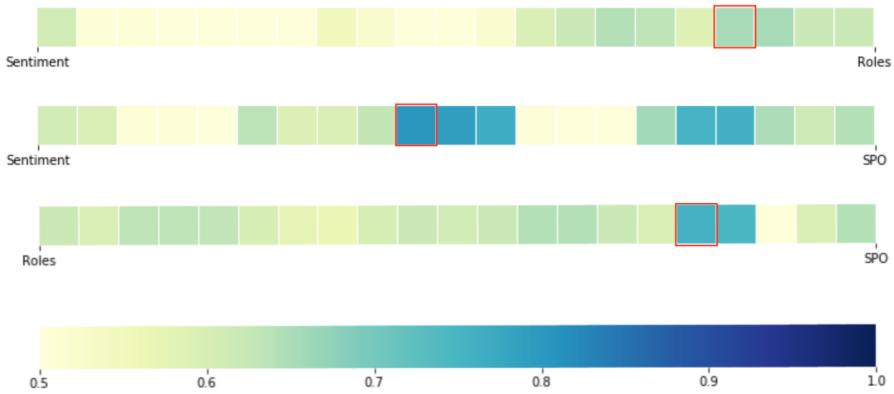


Figure 6: F1-score distribution over modalities weights for Corpus 1

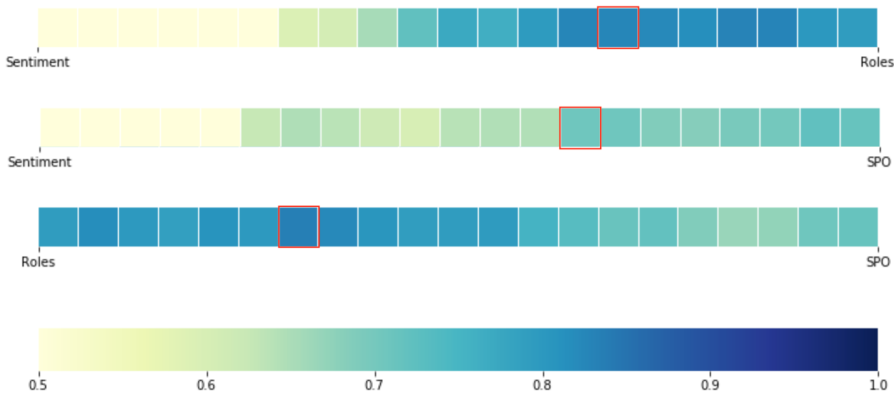


Figure 7: F1-score distribution over modalities weights for Corpus 2

6. Conclusion and Future work

We presented a model for opinion mining in Russian texts that was able to perform two-class clusterization with the F1-score above 0.85 on two datasets. We presented algorithms for mining three syntactic and semantic features: SPO triples, Fillmore roles and sentiment lexicon and compared them with a baseline lexical model. The proposed probabilistic model was trained on all three features as well as their combinations. We demonstrated that combining syntactic and semantic features allows to classify opinionated texts accurately even when single-feature models show low quality and provides a significant advantage over the baseline. We collected and labelled two collections of news on political events to conduct the experiments and made them available for public use.

In this work, we did not determine what weights the features should be combined with to provide optimal clustering quality. To put it in simple terms, we proved that an opinion is defined by mentioned facts, sentiment lexicon and semantic roles of the words used but did not determine “how much” each feature contributed to expressing an opinion. Thus, applying the algorithm at its current state on a raw corpus requires some labelled data to optimise the weights. Adapting self-learning techniques requires widening existing datasets and collecting new ones which we plan to do in the future.

7. Acknowledgements

This work is supported by Russian Foundation for Basic Research, grant 20-07-00936.

References

1. *Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai*. “Topic sentiment mixture: Modeling facets and opinions in weblogs”. In Proceedings of the World Wide Conference (2007), pp. 171–180.
2. *B. Pang, L. Lee*. “Opinion Mining and Sentiment Analysis. Foundations and Trends”. In Information Retrieval (2008), pp. 1–135.
3. *M. S. Hajmohammadi, R. Ibrahim, Z. A. Othman* “Opinion Mining and Sentiment Analysis: A Survey”. In International Journal of Computers & Technology Vol. 2, No. 3 (2012).
4. *M. J. Paul, R. Girju* “Cross-Cultural Analysis of Blogs and Forums with Mixed-Collection Topic Models”. In Proc. of EMNLP ’09 (2009), pp. 1408–1417.
5. *Y. Fang, L. Si, N. Somasundaram, Z. Yu* “Mining Contrastive Opinions on Political Texts using Cross-Perspective Topic Model”. In: Proc. of WSDM ’12 (2012), pp. 63–72.
6. *R. Balasubramanyan, W. W. Cohen, D. Pierce, D. P. Redlawsk* “Modeling Polarizing Topics: When Do Different Political Communities Respond Differently to the Same News?”. In: Proc. of the Sixth International AAAI Conference on Weblogs and Social Media (2012), pp. 18–25.
7. *M. J. Paul, R. Girju* “A Two-Dimensional Topic-Aspect Model for Discovering Multi-Faceted Topics”. In: Proc. of AAAI ’10 (2010), pp. 545–550.
8. *X. Zhu, D. Klabjan, P. N. Bless* “Unsupervised Terminological Ontology Learning based on Hierarchical Topic Modeling”. In: Proc. of ACL 17 (2017).
9. *E. I. Bolshakova, K. V. Vorontsov and others* “Automatic word processing in natural language and data analysis”. pp. 195–228.
10. *Koltsova, O. Yu, S. Alexeeva and S. Kolcov* “An opinion word lexicon and a training dataset for russian sentiment analysis of social media.”. In: Proc. of the International Conference “Dialogue 2016” (2016).
11. *Lashevich G. et al.* “Creating Russian WordNet by Conversion.”. In: Proc. of the International Conference “Dialogue 2016” (2016).

12. *P. Sobkowicz, M. Kaschesky, G. Bouchard* “Opinion mining in social media: Modeling, simulating, and forecasting political opinions in the web”. In: *Government Information Quarterly* vol. 29 (2012) pp. 470–479.
13. *Charles J. Fillmore* “Some problems for case grammar”. In: *22nd Annual Round Table. Linguistics: Developments of the Sixties Viewpoints of the Seventies. Volume 24 of Monograph Series on Language and Linguistics.* (1971) pp. 35–56.
14. *H. Wang and C. Zhai* “Generative Models for Sentiment Analysis and Opinion Mining”. In: *Springer International Publishing AG* (2017).
15. *S. M. Kim, E. Hovy* “Extracting opinions, opinion holders, and topics expressed in online news media text”. In: *Proc. of the Workshop on Sentiment and Subjectivity in Text* (2006) pp. 1–8.
16. *K. Moilanen, S. Pulman* “Sentiment Composition”. In: *Proc. of RANLP-2007* (2007) pp. 378–382.
17. *I. A. Karpov, M. V. Kozhevnikov, V. I. Kazorin, N. R. Nemov* “Entity Based sentiment analysis using syntax patterns and convolutional neural network”. In: *Proc. of the International Conference “Dialogue 2016”* (2016) pp. 378–382.
18. *A. O. Shelmanov, D. A. Devyatkin* “Semantic role labeling with neural networks for texts in Russian”. In: *Proceedings of the International Conference “Dialogue 2017”* (2017).
19. *D. Marcheggiani, A. Frolov, and I. Titov* “A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling”. In: *arXiv preprint arXiv:1701.0259* (2017).
20. *C. A. Thompson, R. Levy, and C. D. Manning* “A Generative Model for Semantic Role Labeling”. In: *Springer-Verlag Berlin Heidelberg 2003* (2003), pp. 397–408.
21. *A. Giuglea and A. Moschitti* “Semantic Role Labeling via FrameNet, VerbNet and PropBank”. In: *Proc. of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL* (2006), pp. 929–936.