Automatic extraction of relevant keyphrases for the study of issue competition

Miguel Won, Bruno Martins and Filipa Raimundo

April 4, 2019

Automatic extraction of relevant keyphrases for the study of issue competition.

Miguel Won[1], Bruno Martins[1], and Filipa Raimundo[2]

[1] INESC-ID, University of Lisbon
miguel.won@tecnico.ulisboa.pt
[2] ICS, University of Lisbon

Abstract. The use of keyphrases is a common oratory technique in political discourse, and politicians often guide their statements by recurrently making use of keyphrases. We propose a statistical method for extracting keyphrases at document level, combining simple heuristic rules. We show that our approach can compete with state-of-the-art systems. The method is particularly useful for the study of policy preferences and issue competition, which relies primarily on the analysis of political statements contained in party manifestos and speeches. As a case study, we show an analysis of Portuguese parliamentary debates. We extract the most used keyphrases from each parliamentary group speech collection to detect political issue emphasis. We additionally show how keyphrase clouds can be used as visualization aids to summarize the main addressed political issues.

Keywords: Keyphrase Extraction · Political discourse · Information Retrieval .

1 Introduction

In recent years, the study of policy preferences and issue competition has increased considerably [26,27,38,16,17,42,43,22]. To identify what issues political actors emphasize or de-emphasize and what their policy priorities are, scholars have engaged in collecting and coding large quantities of text, from speeches to manifestos [17,42,43].

The assumption is that political actors' statements are a more accurate account of where they stand on a particular issue than actual behavior. With such growing interest in issue competition and policy preferences, several methods have been developed to analyze large amounts of qualitative data systematically. [16,17,42]. Traditionally, these works rely upon human annotated text data, in particular, words and sentences (or quasi-sentences) coded into issue categories. One example vastly used by scholars is the Comparative Manifesto Project (CMP) [3] that makes available party manifestos annotated into a restrict code system, which informs about political issue addressing within the texts.

[3] https://manifesto-project.wzb.eu

One problem with this type of data is that party manifestos are limited in time and written once every election. Studies about internal dynamics of each parliamentary groups during the legislative term, party reaction to relevant political events or the constant action/reaction of the party agenda with the media own agenda, are not possible to be developed using the party manifestos only. In [17] the authors overcome this problem using parliamentary activities, such as questions to the government and parliamentary debates. However, the authors had to implement a coding system and perform human codding. For larger parliaments, with a high volume of daily activity, this manual annotation can be highly costly.

Politicians often guide their statements by recurrently making use of keyphrases. The wise use of keyphrases is a common oratory technique in political discourse that allows political actors to transmit the most important ideas associated with their political position. The identification of relevant keyphrases used in political texts can help to synthetically frame political positions and identify the relevant topics addressed by a politician or political group agenda. With today's massive electronic and public data, such as parliament debates, press releases, electoral manifestos or news media articles, several studies focused on the development of statistical algorithms that extract relevant information from large-scale political text collections [18].

In the present work, we propose to use the keyphrase framework to identify attention given by politicians. We offer a simple method that automatically extracts the most relevant keyphrases associated with a political text, such as given speech, a public statement or an opinion article. We show that using a combination of simple text statistical features is possible to achieve results that compete with alternative state-of-the-art methodologies. We evaluate the method performance with annotated corpora typical used to assess the Natural Language Processing task of Keyphrase Extraction. We additionally show a case study where we apply the proposed methodology to a corpus composed by a collection of speeches given during the plenary sessions of the Portuguese Parliament and propose a visualization scheme using a word cloud.

2 Related work

Automatic extraction of keyphrases is usually divided into two main branches: supervised and unsupervised. Within the unsupervised approaches, there are two main frameworks commonly used: graph-based and topic cluster. The first approach was originally proposed with TextRank algorithm [33]. TextRank generates a graph form a text document, where each node is a word and each edge a co-occurrence. Centrality measures, such as PageRank [36], are used to score each word and build a ranking system of keyphrases. More recent graph methods are SingleRank [44] that weight the graph edges with the number of co-occurrences, and SGRank [12] that combines several statistical features to weight the edges between candidates of keyphrases. Regarding topic clusters, methods such as KeyCluster [29] and CommunityCluster [19] cluster candidates to keyphrases semantically similar, resulting in topic clusters. TopicRank [8] is also a relevant work, by joining these two approaches and generating a graph with topic clusters. More recent works use word embeddings framework [34]. One example is [45] where the authors use a graph-based approach and weight the edges with the semantic word embeddings distances. In a more recent work [3] it is proposed to use the semantic distance between the full document and the keyphrases representation, using Sent2Vec [37]. This latter method, named by the authors as EmbedRank, represents the current state-of-the-art performance for extraction systems.

Supervised methodologies use annotated data to train statistical classifiers using syntactic features, such as part-of-speech tags, tf-idf and first position [23,15,35]. Also, works such as [31,30] incorporate additionally external semantic information extracted from Wikipedia.

All these methods extract keyphrases present in the texts. Nevertheless, datasets commonly used to evaluate keyphrase extraction systems contain human annotated keyphrases not present in the documents. Some recently some works have dedicated special attention to generative models, where the extracted keyphrases are not necessarily present in the texts, but generated on-the-fly. Examples of works are [10,32], where the authors use a neural networks deep learning framework to generate relevant keyphrases. For some annotated datasets, the performance of these systems improved the current state-of-art results for extraction systems. We note that in the present work we propose a method that extracts keyphrases present only in the input text.

In respect to political issues analysis, recent works have intensely used CMP framework to signal the political issues from the analyzed corpora. In [16] the authors use CMP to analyze several Western European party manifestos since 1950 to study how issue priorities evolved with time. In [17] the authors propose a model to frame the issue competition between parliamentary groups, using Danish parliament activity data; and in [42] questions from Belgium and Denmark MPs are analyzed to study the issue emphasis dynamics between the MPs.

3 Keyphrases extraction

In the same line of previous works in automatic keyphrases extraction, we follow a three-step process [21]: first, we identify a set of potential candidates to keyphrases; second, we calculate a score for each candidate; and third, we select the top-ranked candidates.

3.1 Candidate identification

The first step to select the best potential candidates to relevant keyphrases is to use morphosyntactic patterns. The use of part-of-speech (POS) filters is a common practice in this type of task, where traditionally only nouns and adjectives are filtered in [33,44,28]. Works such as [1,35] impose additionally morphosyntactic rules for candidates, e.g., the candidate must be a noun-phrase or end with a noun. The idea behind these rules comes from the fact that we are searching for keyphrases that represent entities, which are very likely to be expressed by at least one noun. The further addition of adjectives will allow the inclusion of a noun quantifier, allowing the identification of candidates such as "National Health System" (adjective+noun+noun). Following this example, we propose the use of the pattern of at least one noun possible preceded by adjectives [4]. Additionally, to avoid candidates overlap, as well as limit the generation of a high number of candidates, we propose the use of a chunking rule based in the morphosyntactic pattern [5].

In the present study, we also work with Portuguese texts. For this reason, when working with this corpus, we have applied an equivalent pattern but with the appropriate noun/adjective order for Portuguese, as well the possibility to include a preposition [6].

In respect to prepossessing, we apply standard text cleaning procedures by removing candidates containing stopwords, punctuation or numerical digits. For datasets with longer documents, we requested a minimum of two occurrences, where we have considered the stemmed versions of each candidate. We used NLTK Porter Stemmer [7] for English and NLTK RSLP Stemmer for Portuguese [8].

3.2 Candidate scoring

The next step in the pipeline is to estimate the score of each candidate. We show in this work that a scoring system based on the selection of simple heuristic rules can result in state-of-the-art performances. The selection process of such features considered their good performance in previous works. For each candidate we calculate the following features:

- **Term Frequency (*tf*)**: Keyphrases are usually associated with frequent usage [46,25]. Contrary to the common practice that measures each candidate document frequency, we propose to use instead the sum of each word frequency from that constitutes the candidate. We have observed from our experiments that such feature results in better performances.
- **Inverse Document Frequency (*idf*)**: In case the keyphrase extraction task has access to a context corpus, is possible to use additional information, by using the idf metric [40]. Since this feature uses a context corpus, we have evaluated two systems, H_1 and H_2, where only the former considers *idf*.
- **Relative First Occurrence (*rfo*)**: Keyphrase extraction systems frequently use the position of the first occurrence as a statistical feature. It is a result

[4] We use the POS pattern: <ADJ>* <NOUN>+

[5] In our experiments we used NLTK chunker: `https://www.nltk.org/api/nltk.chunk.html`

[6] We use the chunker POS pattern: (<NOUN>+ <ADJ>* <PREP>*)? <NOUN>+ <ADJ>*

[7] `https://www.nltk.org/_modules/nltk/stem/porter.html`

[8] `https://www.nltk.org/_modules/nltk/stem/rslp.html`

from that fact that often relevant keyphrases are used at the beginning of the text [23,35,14,12]. In this work, we use the likeliness that a candidate shows earlier than a randomly sampled phrase of the same frequency. We calculate the cumulative probability of the type $(1 - a)^k$, where $a \in [0, 1[$ measures the position of the first occurrence and k the candidate frequency [11].

- **Length (*len*)**: The candidate size, i.e., the number of words that compose it, can also hint about the candidate likeliness to be a keyphrase. Human readers tend to identify keyphrases with sizes beyond the unigrams, specially bigrams [11,12]. However, a linear score based in the candidate length would result in overweight of lengthy candidates, such as 3 and 4-grams. Therefore, and based on our trials, we propose a simple rule that scores 1 for unigrams and 2 for the remaining sizes.

The final score of each candidate is the result of the product of these four features. From the results shown in the next section, we will see that with these simple four heuristic rules is possible to obtain results that compete with the state-of-the-art. We show the formal description of each feature is equations (1)-(4), and in equations (5)-(6) the two final score systems for a candidate $w = w_1...w_2$ and a set of documents $\{d \in D\}$, with size $N = |D|$. Model H_1 does not take into consideration idf, and therefore it is calculated at document level only. System H_2 includes idf weight that in our experiments is measured by the respective corpus.

$$\mathit{tf}(w_1...w_n) = \sum_{i=1}^{n} \mathit{fr}(w_i, d) \tag{1}$$

$$\mathit{tf\text{-}idf}(w_1...w_n) = \sum_{i=1}^{n} \mathit{fr}(w_i, d) \times \log(\frac{N}{1 + |d \in D : w_i \in d|} \tag{2}$$

$$\mathit{rfo}(w_1...w_n) = (1 - a)^{\mathit{fr}(w_1...w_n, d)} \tag{3}$$

$$\mathit{len}(w_1...w_n) = \begin{cases} 1 : n = 1 \\ 2 : n \geq 2 \end{cases} \tag{4}$$

$$H_1 = \mathit{tf} \times \mathit{rfo} \times \mathit{len} \tag{5}$$

$$H_2 = \mathit{tf\text{-}idf} \times \mathit{rfo} \times \mathit{len} \tag{6}$$

3.3 Top n-rank candidates

After the scoring step, we select the best ranking candidates. It is common to evaluate keyphrase extraction systems at three top-ranking scenarios: the first 5, 10 and 15 candidates. However, the best raking can be influenced by the document size [9]. Smaller documents such as abstracts are likely to contain fewer keyphrases than full-length articles. Table 1 shows that larger document datasets

[9] To measure the document size we use the total number of tokens.

are associated with a higher number of keyphrases per document. Therefore, we propose to extract the n-top ranked candidates dynamically by considering the respective document size. We use the identity shown in the equation (7) to calculate the n-top candidates for each document. We propose a logarithm growth to prevent significant and fast discrepancies between a small text such as an abstract and a full paper. We found the 2.5 parameter by experiment, and therefore one should note that it can be a result of overfitting. However, as we will see below, this dynamical ranking system returns consistent good results for all datasets, which is an indicator that is document size independent.

$$n_{\text{keys}} = 2.5 \times \log_{10}(\text{doc size}) \tag{7}$$

4 Experiments

4.1 Evaluation metric

We follow the standard procedure to evaluate keyphrase extration systems: we measure the macro-average *F-score* (average the *F-score* for each document), using the harmonic mean of *Precision* and *Recall*. For *Precision*, we use the identify $P = \frac{\#\text{correct keyphrases}}{\#\text{extracted keyphrases}}$ and for Recall $R = \frac{\#\text{correct keyphrases}}{\#\text{gold keyphrases}}$. Both *Precision* and *Recall* are calculated using the exact match of the stemmed versions of the extracted keyphrases.

4.2 Datasets

We use five human annotated datasets, four with English texts and a fifth with Portuguese (European). The English datasets are popular sets used to evaluate keyphrase extraction systems. The dataset of Portuguese texts enables us to assess performance with a language rarely tested in keyphrase extraction, as well to validate the case study shown in the present work. Follows a brief description of each dataset:

- **Inspec** [23] is a dataset with 2000 scientific journal abstracts. Equivalently to previous works [33,44] we evaluate the performance using the test dataset, which contains a total of 500 documents. The dataset contains three files per abstract, one containing the text and the remaining two the controlled and uncontrolled keyphrases, separately. We have used the uncontrolled set of keyphrases for evaluation.
- **DUC2001** [44] is a corpus with 308 newspapers articles. We have used the set of human-annotated keyphrases for each document available with the dataset. In our experiments, this dataset is the only English corpus made from non-academic texts.
- **Semeval 2010** [24] is a common dataset used in automatic keyphrases extraction research. It contains scientific articles with author and reader annotated keyphrases. We have used the test dataset only, made of 100 papers, and the combined set of keyphrases.

– **Nguyen** [35] contains 211 scientific conference papers with author and reader annotated keyphrases. We considered all articles and the combined set of keyphrases.
– **Geringonça** is a new dataset of news articles written in Portuguese and extracted from the political news web portal `http://geringonca.com/`. The dataset contains a total of 800 pieces, and for each article, a set of keyphrases were assigned by the respective authors.

Table 1. Datasets descriptive statistics. From the second column left we show: the total number of documents used for evaluation; average tokens per document; the average number of keyphrases per document; maximum possible recall for each dataset.

Dataset	No. docs	Avg tok	Keys/Doc	Max Recall
Inspec	500	139.49	9.81	78.20
DUC2001	308	896.78	8.06	94.95
Semeval 2010	100	2147.26	14.43	77.36
Nguyen	209	8777.70	10.86	84.58
Geringonça	894	284.10	5.13	84.70

We show in Table 1 some descriptive statistics of the datasets. SemEval and Ngyuen datasets contain large documents with a high number of gold keyphrases.

In Fig. 1 we show the distribution of keyphrases size for each dataset. For the English corpora, we confirm that keyphrases sizes go beyond unigrams, especially to bigrams [11]. We don't see the same type of distribution for the Portuguese dataset. However, we note that the annotation process of this dataset was not supervised nor thought to be used as a gold standard to keyphrase extraction systems. It was created during the writing process of the news pieces and likely annotated as a contribution to a framework of keywords/tags used by the web portal.

A known problem when working with the datasets summarized in Table 1 is the non-presence of gold keyphrases in the respective document. For this reason, we show in Table 1 the maximum recall that any system can achieve. Equivalently to others works [20], we include these missing keyphrases in our gold sets.

In respect to POS prepossessing, all English datasets, except SemEval, were processed with Stanford POS tagger [41]. For the SemEval we used the prepossessed dataset available in `https://github.com/boudinfl/semeval-2010-pre` [7]. For the Portuguese Geringonça dataset, all documents were processed by the POS tagger for Portuguese, LX-Tagger [9]. All other tasks related to text processing, such as word and sentence tokenization were performed using NLTK [10].

[10] `https://www.nltk.org/`

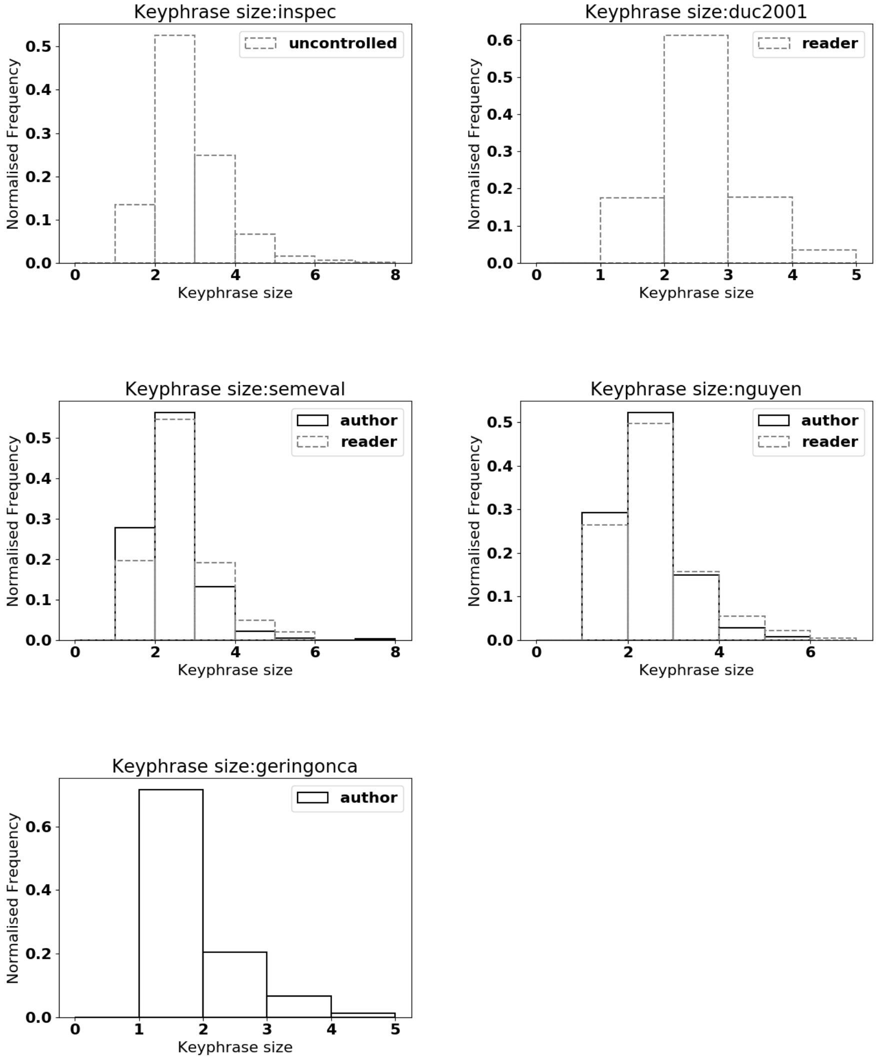

Fig. 1. Distributions of keyphrase sizes (# of words that compose the keyphrase) for each dataset. The dark line (continuous) histograms show the distribution resulted when keyphrases were annotated by the author, and grey line (dashed) histograms when the keyphrases were annotated by readers.

4.3 Results

We show in Table 2 the resulted F-scores for the English corpora. We name our method KCRank, where H1 and H2 refer to the use of the respective context dataset corpus (equation (6)). Detailed results, in particular, Precision and Recall, are shown in Table 4 in the appendix section. For comparison reasons we also show the F-score when considering *tf-idf* as a baseline, as well four alternative state-of-the-art methods: KPMiner [13], TopicRank [8], MultipartitieRank [6] and EmbededRank [3]. We have used pke tool [5] to extract keyphrases with KPMiner, TopicRank and MultipartitieRank methods. For EmbededRank (s2v version) we used our own implementation code [11].

From Table 2 we show that, except for NUS dataset, KCRank H1 and KCRank H2 models return consistent better F-score results. We also show that there is no significant difference between KCRank H1 and KCRank H2 performance. This result indicates that the use of context corpus does not significantly increase the overall performance. In respect the alternative methods, only EmbedRank returns a competitive F-score. For the NUS dataset, KPMiner shows the best results. We note that KPMiner relies in similar principles we used to build KCRank method, namely the *tf-idf*. Table 2 shows that KPMiner results are very similar to KCRank for SemEval and NUS datasets, but not for Inspec and DUC. This is an indication that KPMiner method may be fitted to work with long scientific papers only.

In Table 3 we show the F-scores for the Geringonça dataset, and detailed results are shown in Table 5 in the appendix section. For this experiment, the KPMiner model gets the best results, surprisingly followed by the baseline tf-idf. One possible reason for these results is the different distribution of the size of keyphrases that we observe for the Geringonça dataset when compared to the English datasets. From Fig. 1 we see that most keyphrases have size one, which is not the case with the English datasets. This statistical difference results in a negative contribution by the keyphrase length feature, used by KCRank. To test the impact of this feature, we conducted an experiment where we turned off the keyphrase length feature. We show the results in Table 3, where we have identified this modified version of KCRank by KCRank$_{\text{length}}$. The higher performance when this feature is turned off, confirms that the keyphrase size feature is contributing negatively. As previously pointed out, the Gerigonça dataset was created as part of a web portal's tag system. We claim that this process skewed the annotation process that resulted in the predominance of unigrams.

[11] For the English datasets, the keyphrase extration was performed with pre-trained embeddings for unigrams available in `https://github.com/epfml/sent2vec`. For Portuguese, we generated fastText [4] embeddings using a compilation of all sentences present in the Gerigonça dataset and the parliamentary speeches used in the case study shown in section 5.

Table 2. F-scores of KCRank and comparison with state-of-the-art systems, when using the English language datasets. The results for Semeval and NUS datasets were obtained with an additional filter where candidates with less than two occurrences were excluded.

N	Method	Inspec	DUC	Semeval (min=2)	NUS (min=2)
5	tfidf	29.04	17.28	14.39	15.74
	KPMiner	16.50	11.69	**18.06**	**21.13**
	TopicRank	23.30	17.59	11.69	12.97
	MultipartiteRank	23.73	18.54	12.68	13.50
	EmbedRank	29.38	25.77	12.17	12.10
	KCRank:H1	33.70	**27.39**	16.22	12.41
	KCRank:H2	**34.56**	27.10	16.57	13.30
10	tfidf	36.53	20.75	18.40	19.93
	KPMiner	19.65	14.49	21.53	**24.92**
	TopicRank	26.33	19.43	14.07	15.10
	MultipartiteRank	27.57	21.64	14.54	16.93
	EmbedRank	37.10	29.56	17.39	15.44
	KCRank:H1	39.22	**31.00**	20.60	16.16
	KCRank:H2	**39.62**	30.57	**22.05**	17.11
15	tfidf	37.81	20.81	20.07	20.17
	KPMiner	20.55	14.75	22.44	**24.44**
	TopicRank	26.93	19.36	14.43	14.20
	MultipartiteRank	28.63	22.09	15.70	16.31
	EmbedRank	38.55	29.30	20.27	17.38
	KCRank:H1	38.53	30.55	22.32	17.13
	KCRank:H2	**39.14**	**30.77**	**22.66**	17.63
dynamic	tfidf	37.29	20.92	20.03	18.88
	KPMiner	20.18	14.80	21.39	**22.51**
	TopicRank	26.69	19.16	13.76	12.97
	MultipartiteRank	28.46	22.08	16.20	15.77
	EmbedRank	37.83	29.45	20.14	16.24
	KCRank:H1	39.40	30.40	21.80	16.71
	KCRank:H2	**39.62**	**30.58**	**22.28**	17.84

Table 3. F-scores of KCRank and comparison with state-of-the-art systems, when using the Geringonça datasets. The F-scores results of an experiment using KCRank system with keyphrase size feature turned off is also shown (in light bold).

N	Method	Geringonça
5	tfidf	26.29
	KPMiner	**25.40**
	TopicRank	13.00
	MultipartiteRank	13.58
	EmbedRank	18.57
	KCRank:H1	10.27
	KCRank:H2	14.49
	KCRank$_{length}$:H1	**13.21**
	KCRank$_{length}$:H2	**21.01**
10	tfidf	21.58
	KPMiner	**25.08**
	TopicRank	14.72
	MultipartiteRank	15.34
	EmbedRank	18.87
	KCRank:H1	14.44
	KCRank:H2	18.50
	KCRank$_{length}$:H1	**18.37**
	KCRank$_{length}$:H2	**22.79**
15	tfidf	18.65
	KPMiner	**22.51**
	TopicRank	15.05
	MultipartiteRank	15.43
	EmbedRank	17.89
	KCRank:H1	17.68
	KCRank:H2	19.72
	KCRank$_{length}$:H1	**19.98**
	KCRank$_{length}$:H2	**21.06**
dynamic	tfidf	20.23
	KPMiner	**24.00**
	TopicRank	14.87
	MultipartiteRank	15.35
	EmbedRank	18.53
	KCRank:H1	16.19
	KCRank:H2	19.21
	KCRank$_{length}$:H1	**19.09**
	KCRank$_{length}$:H2	**22.49**

5 Key-phrase extraction using Portuguese parliamentary debates

Like other national parliaments, the Portuguese Parliament produces faithful transcripts of the speeches given in the plenary sessions and makes them publicly available in electronic format. For our study, we collected transcriptions from the Portuguese Parliament website[12], referring to the last complete legislative term (i.e., from June 2011 to October 2015), together with information on each member of parliament (MP). During this period, the chamber was composed with MPs from five different parties: The Greens (PEV), the Portuguese Communist Party (PCP), the Left Block (BE), the Socialist Party (PS), the Social Democratic Party (PSD) and the Social Democratic Centre (CDS-PP). In total, we have collected 16,993 speeches.

Using the keyphrase extraction method described in this work, we can identify the most central and recurrent political issues addressed during the plenary debates. The keyphrase identification of each speech can reveal the political priorities of each parliamentary group and hint about their expressed agenda. Therefore, we extracted the keyphrases from each speech, where we used model KCRank H2 with context corpus the respective parliamentary group collection.

5.1 Candidates Selection

We observed that the extraction of clean and intelligent keyphrases from the speech dataset is a difficult task in this particular corpora, due to the many repetitive words and expressions used by the MPs. Typical examples are expressions such as "Mr. President", "party" and "draft bill" that due to their frequent use in this type of text are scored as relevant keyphrases. Consequently, for the candidate selection step, we processed all speeches transcriptions with the pipeline used for Geringonça dataset with two additional filters: a minimum of 5 occurrences criteria and an extension of the stopwords list with these common words and expressions [13].

5.2 Visualisation

As visualization scheme, we propose the use of word clouds, with general guidelines introduced in previous studies [2,39], regarding the choice of visual variables and spatial layout (e.g., we preferred a circular layout that tends to place the most relevant key-phrases in the center). Each keyphrase cloud summarizes the number of occurrences of each keyphrase in the dataset analyzed, i.e., the number of speeches in which the respective candidate was selected as a keyphrase. We use this metric to encode the keyphrase font-size and color, where the darker color represents the keyphrase with the highest number of occurrences. With such visualization aid, the reader can obtain, in a dense image, a high number

[12] `http://debates.parlamento.pt`

[13] We manually annotated a list of approximately 100 terms.

of relevant keyphrases from the text collection, and therefore better capture the main issues addressed.

Fig. 2 shows a keyphrase cloud for the top 40 key-phrases extracted from the collection composed of PSD speeches[14].

The two most relevant issues addressed by this text collection are "european union" and "memorandum of understanding". The use of this terms is possibly related to the fact that PSD was a government support party (in coalition with CDS-PP) during the considered legislature, and was responsible for implementing Troika austerity measures. In Fig. 7 we show the equivalent keyphrase cloud for the CDS-PP speeches, from which we can see a very similar pattern. Fig. 2 also shows a mix of topics related with economic affairs, such as "work", "economic growth" or "job creation", and welfare state related, such as "social security" and "national health service".

Fig. 3 shows the equivalent keyphrase cloud but when considering the collection of speeches from PS, major opposition party during the same legislature. The cloud shows a different scenario, with PS speeches emphasizing issues related with the public sector, namely with the keyphrases "national health service", "social security", "public services". It also shows how PS emphasized "constitutional court" (during the legislative term several austerity measures were requested to be audited by the Portuguese Constitutional Court) and "tax increase". Both PSD and PS clouds show relevant false positives keyphrases such as "theme", "situation" and "day". This unwanted effect is stronger in the PSD cloud. One possible explanation for the stronger effect of false positives in PSD could be the distribution of PSD speeches by more different political issues. Being the main government support party it needs to be more responsive to the different opposition parties agendas [17]. Such an effect will result that a high number of speeches will be not be used as agenda-setting but as a response to the opposition parties questions during the plenary debates.

We show in the appendix in Fig. 4-7 the equivalent results when considering the text collections from the remaining parties. The results show that keyphrases such as "european union", "national health service" and "social security" are present in all word clouds, indicating their relevance as a top issues.

Finally, we note that the use of keyphrases beyond unigrams, allows us to identify with more precision the relevant political issues. Had the analysis been based in unigrams only, the identification of lengthy and more specific keyphrases as relevant issues would be lost. The use of n-grams is an essential advantage of the present method since it allows the political scientist to keep track of the many political issues related entities with n-gram size.

[14] All keyphrases were translated from Portuguese.

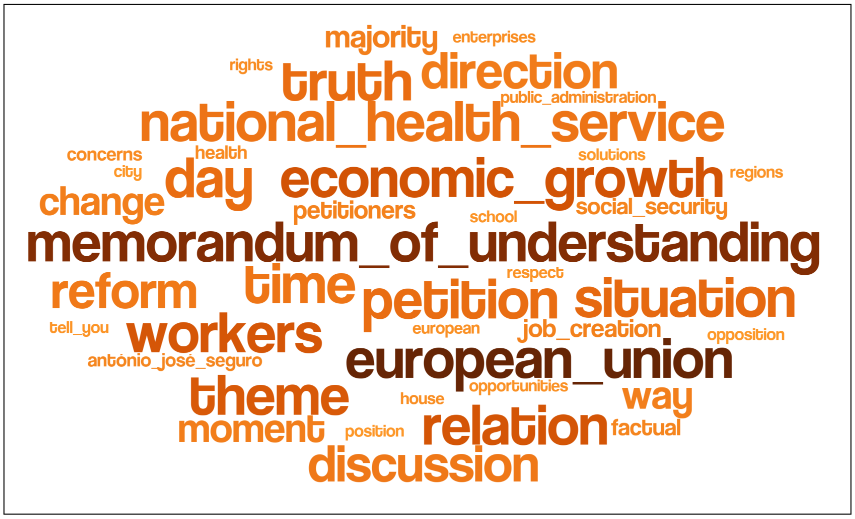

Fig. 2. The 40 most relevant keyphrases extracted from PSD speeches collection.

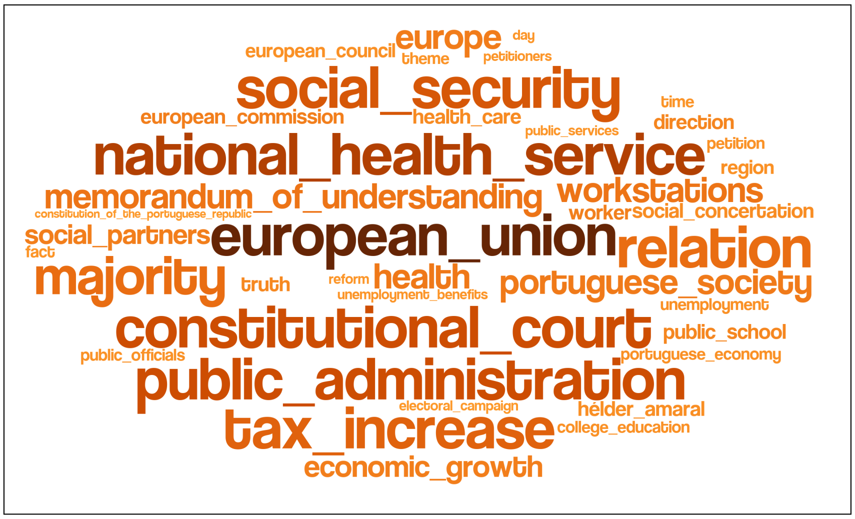

Fig. 3. The 40 most relevant keyphrases extracted from PS speeches collection.

6 Conclusion

We present a method that automatically extracts keyphrases from text documents. We combine simple statistical features to generate a ranking list of candidates to keyphrases. From this list, we propose a dynamic selection of the top candidates, based in the document length. We test our methodology with different datasets, commonly used to evaluate keyphrase extraction systems, and show that the proposed method competes with state-of-the-art alternatives. Due to its simplicity, the proposed method can be implemented in any lightweight software application or web application to extract keyphrases from text documents, at document level and *on the fly*.

We show a small case study using plenary speeches given at the Portuguese parliament. From the proposed methodology we extract the most relevant keyphrases from each parliamentary group set of speeches and construct keyphrases clouds. We show how such clouds can efficiently summarize issue agenda-setting by the respective parties. Furthermore, we show that using keyphrases that go beyond simple unigrams allows a higher precision identification of entities of interest.

Acknowledgement.

This work was supported by Fundação para a Ciência e Tecnologia (FCT), SFRH/BPD/104176/2014 to M. W., SFRH/BPD/86702/2012 to F.R., and FCT funding POCI/01/0145/FEDER/031460 and UID/CEC/50021/2019. We also want to thank the `http://geringonca.com/` team for having made their data available.

References

1. Barker, K., Cornacchia, N.: Using noun phrase heads to extract document keyphrases. In: Conference of the Canadian Society for Computational Studies of Intelligence. pp. 40–52. Springer (2000)
2. Bateman, S., Gutwin, C., Nacenta, M.: Seeing things in the clouds: The effect of visual features on tag cloud selections. In: Proceedings of the ACM Conference on Hypertext and Hypermedia (2008)
3. Bennani-Smires, K., Musat, C., Jaggi, M., Hossmann, A., Baeriswyl, M.: Embedrank: Unsupervised keyphrase extraction using sentence embeddings. arXiv preprint arXiv:1801.04470 (2018)
4. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
5. Boudin, F.: pke: an open source python-based keyphrase extraction toolkit. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations. pp. 69–73. The COLING 2016 Organizing Committee, Osaka, Japan (December 2016), `http://aclweb.org/anthology/C16-2015`
6. Boudin, F.: Unsupervised keyphrase extraction with multipartite graphs. arXiv preprint arXiv:1803.08721 (2018)
7. Boudin, F., Mougard, H., Cram, D.: How document pre-processing affects keyphrase extraction performance. arXiv preprint arXiv:1610.07809 (2016)
8. Bougouin, A., Boudin, F., Daille, B.: Topicrank: Graph-based topic ranking for keyphrase extraction. In: International Joint Conference on Natural Language Processing (IJCNLP). pp. 543–551 (2013)

9. Branco, A., Silva, J.R.: Evaluating solutions for the rapid development of state-of-the-art pos taggers for portuguese. In: LREC (2004)
10. Chen, J., Zhang, X., Wu, Y., Yan, Z., Li, Z.: Keyphrase generation with correlation constraints. arXiv preprint arXiv:1808.07185 (2018)
11. Chuang, J., Manning, C.D., Heer, J.: "Without the clutter of unimportant words": Descriptive keyphrases for text visualization. ACM Transactions on Computer-Human Interaction **19** (2012)
12. Danesh, S., Sumner, T., Martin, J.H.: Sgrank: Combining statistical and graphical methods to improve the state of the art in unsupervised keyphrase extraction. Lexical and Computational Semantics (* SEM 2015) p. 117 (2015)
13. El-Beltagy, S.R., Rafea, A.: Kp-miner: A keyphrase extraction system for english and arabic documents. Information Systems **34**(1), 132–144 (2009)
14. Florescu, C., Caragea, C.: Positionrank: An unsupervised approach to keyphrase extraction from scholarly documents. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). vol. 1, pp. 1105–1115 (2017)
15. Frank, E., Paynter, G.W., Witten, I.H., Gutwin, C., Nevill-Manning, C.G.: Domain-specific keyphrase extraction. In: 16th International joint conference on artificial intelligence (IJCAI 99). vol. 2, pp. 668–673. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999)
16. Green-Pedersen, C.: The growing importance of issue competition: The changing nature of party competition in western europe. Political studies **55**(3), 607–628 (2007)
17. Green-Pedersen, C., Mortensen, P.B.: Who sets the agenda and who responds to it in the danish parliament? a new model of issue competition and agenda-setting. European Journal of Political Research **49**(2), 257–281 (2010)
18. Grimmer, J., Stewart, B.M.: Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political analysis pp. 267–297 (2013)
19. Grineva, M., Grinev, M., Lizorkin, D.: Extracting key terms from noisy and multitheme documents. In: Proceedings of the 18th international conference on World wide web. pp. 661–670. ACM (2009)
20. Hasan, K.S., Ng, V.: Conundrums in unsupervised keyphrase extraction: making sense of the state-of-the-art. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. pp. 365–373. Association for Computational Linguistics (2010)
21. Hasan, K.S., Ng, V.: Automatic keyphrase extraction: A survey of the state of the art. In: ACL (1). pp. 1262–1273 (2014)
22. Hobolt, S.B., De Vries, C.E.: Issue entrepreneurship and multiparty competition. Comparative Political Studies **48**(9), 1159–1185 (2015)
23. Hulth, A.: Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 conference on Empirical methods in natural language processing. pp. 216–223. Association for Computational Linguistics (2003)
24. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In: Proceedings of the 5th International Workshop on Semantic Evaluation. pp. 21–26. Association for Computational Linguistics (2010)
25. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Automatic keyphrase extraction from scientific articles. Language resources and evaluation **47**(3), 723–742 (2013)
26. Klingemann, H.D.: Mapping policy preferences. Oxford University Press (2001)

27. Klingemann, H.D.: Mapping policy preferences II: estimates for parties, electors, and governments in Eastern Europe, European Union, and OECD 1990-2003, vol. 2. Oxford University Press on Demand (2006)
28. Liu, F., Pennell, D., Liu, F., Liu, Y.: Unsupervised approaches for automatic keyword extraction using meeting transcripts. In: Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics. pp. 620–628. Association for Computational Linguistics (2009)
29. Liu, Z., Li, P., Zheng, Y., Sun, M.: Clustering to find exemplar terms for keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1. pp. 257–266. Association for Computational Linguistics (2009)
30. Lopez, P., Romary, L.: Humb: Automatic key term extraction from scientific articles in grobid. In: Proceedings of the 5th international workshop on semantic evaluation. pp. 248–251. Association for Computational Linguistics (2010)
31. Medelyan, O., Frank, E., Witten, I.H.: Human-competitive tagging using automatic keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3. pp. 1318–1327. Association for Computational Linguistics (2009)
32. Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., Chi, Y.: Deep keyphrase generation. arXiv preprint arXiv:1704.06879 (2017)
33. Mihalcea, R., Tarau, P.: TextRank: Bringing order into texts. In: Proceedings of Conference on Empirical Methods on Natural Language Processing (2004)
34. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
35. Nguyen, T.D., Kan, M.Y.: Keyphrase extraction in scientific publications. In: International conference on Asian digital libraries. pp. 317–326. Springer (2007)
36. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Tech. rep., Stanford InfoLab (1999)
37. Pagliardini, M., Gupta, P., Jaggi, M.: Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. In: NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics (2018)
38. Petrocik, J.R.: Issue ownership in presidential elections, with a 1980 case study. American journal of political science pp. 825–850 (1996)
39. Rivadeneira, A.W., Gruen, D.M., Muller, M.J., Millen, D.R.: Getting our head in the clouds: Toward evaluation studies of tagclouds. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (2007)
40. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information processing & management **24**(5), 513–523 (1988)
41. Toutanova, K., Manning, C.D.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13. pp. 63–70. Association for Computational Linguistics (2000)
42. Vliegenthart, R., Walgrave, S.: Content matters: The dynamics of parliamentary questioning in belgium and denmark. Comparative Political Studies **44**(8), 1031–1059 (2011)

43. Wagner, M., Meyer, T.M.: Which issues do parties emphasise? salience strategies and party organisation in multiparty systems. West European Politics **37**(5), 1019–1045 (2014)

44. Wan, X., Xiao, J.: Single document keyphrase extraction using neighborhood knowledge. In: Proceedings of the AAAI Conference on Artificial Intelligence (2008)

45. Wang, R., Liu, W., McDonald, C.: Corpus-independent generic keyphrase extraction using word embedding vectors. In: Software Engineering Research Conference. vol. 39 (2014)

46. Yih, W.t., Goodman, J., Carvalho, V.R.: Finding advertising keywords on web pages. In: Proceedings of the 15th international conference on World Wide Web. pp. 213–222. ACM (2006)

Table 4. Precision, Recall and F-scores for the English datasets of KCRank and comparison with state-of-the-art systems.

N	Method	Inspec			DUC			Semeval (min=2)			NUS (min=2)		
		P	R	F	P	R	F	P	R	F	P	R	F
5	tfidf	37.56	23.67	29.04	21.23	14.57	17.28	27.40	9.76	14.39	22.20	12.19	15.74
	KPMiner	20.64	13.75	16.50	14.61	9.75	11.69	34.20	12.27	18.06	28.71	16.71	21.13
	TopicRank	31.08	18.64	23.30	22.39	14.48	17.59	22.40	7.91	11.69	17.89	10.17	12.97
	MultipartiteRank	31.56	19.01	23.73	23.31	15.40	18.54	24.20	8.59	12.68	18.09	10.77	13.50
	EmbedRank	37.88	23.99	29.38	32.66	21.27	25.77	23.20	8.24	12.17	17.03	9.38	12.10
	KCRank:H1	44.20	27.24	33.70	34.61	22.66	27.39	31.20	10.96	16.22	16.75	9.85	12.41
	KCRank:H2	45.12	28.00	34.56	34.35	22.37	27.10	32.00	11.18	16.57	18.09	10.52	13.30
10	tfidf	33.81	39.72	36.53	18.23	24.09	20.75	22.20	15.71	18.40	19.09	20.85	19.93
	KPMiner	17.62	22.22	19.65	12.60	17.05	14.49	26.00	18.37	21.53	23.73	26.23	24.92
	TopicRank	24.90	27.95	26.33	17.36	22.07	19.43	17.00	12.00	14.07	14.35	15.94	15.10
	MultipartiteRank	25.67	29.78	27.57	19.22	24.76	21.64	17.60	12.38	14.54	16.08	17.88	16.93
	EmbedRank	34.45	40.19	37.10	26.35	33.67	29.56	20.90	14.89	17.39	14.93	15.99	15.44
	KCRank:H1	36.75	42.04	39.22	27.68	35.24	31.00	25.00	17.52	20.60	15.31	17.11	16.16
	KCRank:H2	37.03	42.59	39.62	27.29	34.74	30.57	26.90	18.68	22.05	16.22	18.11	17.11
15	tfidf	30.64	49.38	37.81	15.79	30.52	20.81	19.47	20.70	20.07	16.24	26.61	20.17
	KPMiner	15.77	29.49	20.55	11.05	22.17	14.75	21.87	23.05	22.44	19.55	32.56	24.44
	TopicRank	22.23	34.14	26.93	14.90	27.64	19.36	14.00	14.88	14.43	11.45	18.68	14.20
	MultipartiteRank	23.07	37.74	28.63	16.92	31.81	22.09	15.27	16.15	15.70	13.18	21.40	16.31
	EmbedRank	31.25	50.31	38.55	22.45	42.15	29.30	19.67	20.91	20.27	14.04	22.81	17.38
	KCRank:H1	31.34	49.98	38.53	23.45	43.82	30.55	21.73	22.95	22.32	13.91	22.29	17.13
	KCRank:H2	31.85	50.74	39.14	23.60	44.22	30.77	22.13	23.21	22.66	14.26	23.09	17.63
dynamic	tfidf	32.37	43.97	37.29	15.57	31.87	20.92	17.10	24.18	20.03	13.46	31.59	18.88
	KPMiner	16.82	25.22	20.18	10.87	23.17	14.80	18.30	25.72	21.39	15.96	38.16	22.51
	TopicRank	23.54	30.82	26.69	14.44	28.45	19.16	11.75	16.61	13.76	9.26	21.64	12.97
	MultipartiteRank	24.67	33.61	28.46	16.57	33.09	22.08	13.85	19.51	16.20	11.20	26.60	15.77
	EmbedRank	32.89	44.52	37.83	22.10	44.10	29.45	17.20	24.29	20.14	11.61	27.02	16.24
	KCRank:H1	34.49	45.95	39.40	22.85	45.39	30.40	18.70	26.13	21.80	11.96	27.68	16.71
	KCRank:H2	34.62	46.31	39.62	22.97	45.72	30.58	19.10	26.74	22.28	12.74	29.74	17.84

7 Appendix

Table 5. Precision, Recall and F-scores for the Geringonça datasets of KCRank and comparison with state-of-the-art systems.

N	Method	Geringonça		
		P	R	F
5	tfidf	26.24	26.34	26.29
	KPMiner	25.48	25.33	25.40
	TopicRank	13.09	12.90	13.00
	MultipartiteRank	13.64	13.51	13.58
	EmbedRank	18.37	18.77	18.57
	KCRank:H1	10.00	10.56	10.27
	KCRank:H2	14.30	14.70	14.49
10	tfidf	16.20	32.32	21.58
	KPMiner	18.92	37.19	25.08
	TopicRank	11.35	20.94	14.72
	MultipartiteRank	11.75	22.11	15.34
	EmbedRank	14.07	28.62	18.87
	KCRank:H1	10.68	22.25	14.44
	KCRank:H2	13.75	28.26	18.50
15	tfidf	12.45	37.16	18.65
	KPMiner	15.10	44.26	22.51
	TopicRank	10.81	24.73	15.05
	MultipartiteRank	10.94	26.16	15.43
	EmbedRank	11.89	36.06	17.89
	KCRank:H1	11.71	36.00	17.68
	KCRank:H2	13.09	39.92	19.72
dynamic	tfidf	14.34	34.35	20.23
	KPMiner	17.10	40.25	24.00
	TopicRank	11.02	22.86	14.87
	MultipartiteRank	11.28	24.02	15.35
	EmbedRank	13.06	31.88	18.53
	KCRank:H1	11.35	28.26	16.19
	KCRank:H2	13.51	33.22	19.21

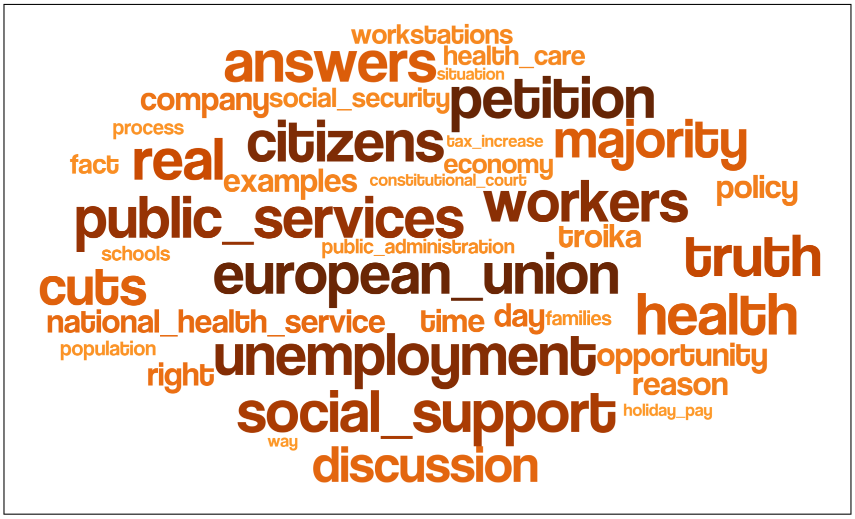

Fig. 4. The 40 most relevant key-phrases extracted from PEV speeches collection.

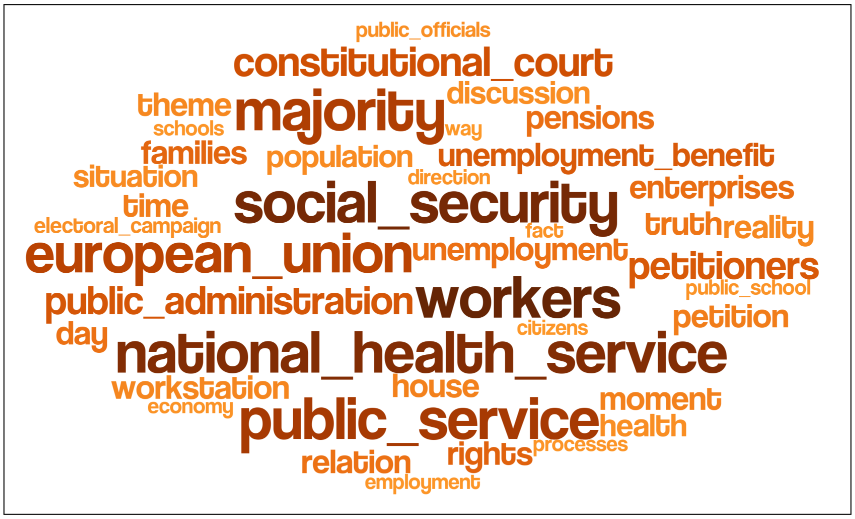

Fig. 5. The 40 most relevant keyphrases extracted from BE speeches collection.

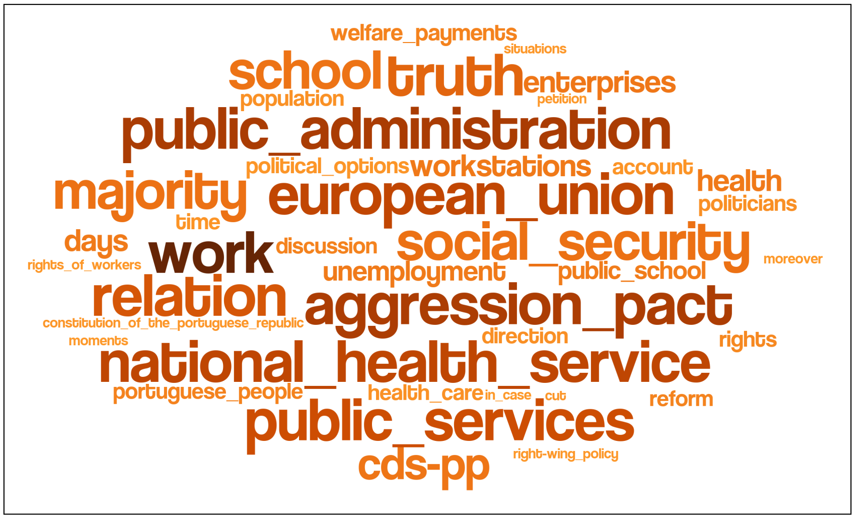

Fig. 6. The 40 most relevant keyphrases extracted from PCP speeches collection.

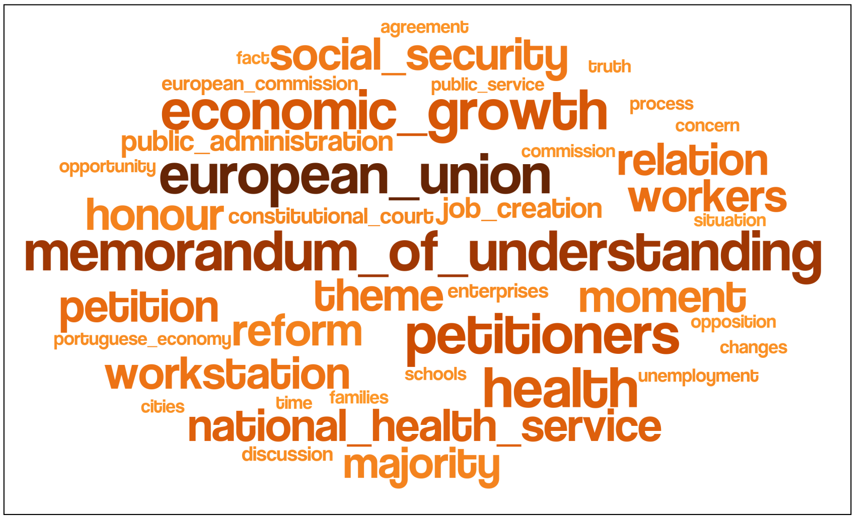

Fig. 7. The 40 most relevant keyphrases extracted from CDS-PP speeches collection.