Automatic Extraction of Quotes and Topics from News Feeds

Luís Sarmento and Sérgio Nunes

Faculdade de Engenharia da Universidade do Porto,
Rua Dr. Roberto Frias, s/n 4200-465 Porto PORTUGAL
{las,ssn}@fe.up.pt

Abstract. The explosive growth in information production poses increasing challenges to consumers, confronted with problems often described as "information overflow". We present `verbatim`, a software system that can be used as a personal information butler to help structure and filter information. We address a small part of the information landscape, namely quotes extraction from portuguese news. This problem includes several challenges, specifically in the areas of information extraction and topic distillation. We present a full description of the problems and our adopted approach. `verbatim` is available online at `http://irlab.fe.up.pt/p/verbatim`.

Key words: Quotes Extraction, News Parsing, Topic Distillation, Named Entity Recognition, Information Extraction, Online Media.

1 Introduction

The current growth in information production poses increasing challenges to consumers, confronted with problems often described as "information overflow". How to deal with such a growing number of information sources? One possible answer to this problem are automatic software tools for information structuring and filtering, bringing some order to an unstructured information landscape. Tools that work as "personal information butlers". We present a contribution to this area with the development of `verbatim`, a system for quotes extraction and classification. `verbatim` acquires information from live news feeds, extracts quotes and topics from news and presents this information in a web-based interface. We choose to investigate the extraction of quotes because it combines several different views over all news topics. Both named entities (e.g. personalities, organizations) and current topics are present when extracting and distilling quotes. Also, we think that such a tool could work as an automatic "watchdog" by confronting quotes by the same entities on the same topics over time. Our main contribution is the deployment of a fully functional prototype, available online at `http://irlab.fe.up.pt/p/verbatim`.

2 Related Work

Quotes extraction from mainstream media sources has been addressed before in academic literature. Pouliquen et al. [6] present a system, named NewsExplorer[1], that detects quotations in multilingual news. The system is able to extract quotes, the name of the entity making the quote and also entities mentioned in the quote. Our work is different since it is focused on a single language (Portuguese) and also addresses the problem of topic extraction and distillation, while most of the related work assumes that news topics have been previously identified. Krestel et al. [5] describe the development of a reported speech extension to the GATE framework. This work is focused on the English language and, contrary to ours, does not address the problem of topic distillation and classification. In [4], the authors propose the TF*PDF algorithm for automatically extracting terms that can be use as descriptive tag. This approach generates a very large set of semantically related terms, but most of them are quite uninformative and innappropriate for being used as a high-level topic tag. Google has recently launched a tool that automatically extracts quotes and topics from online news — In Quotes[2]. The web-based interface is structured in issues (i.e. topics) and displays side-by-side quotes from two actors at a time. However, no implementation details are published about this system.

3 System Overview

The challenge of extracting quotes from live news feeds can be structured in the following generic problems: data acquisition and parsing, quotes extraction, removal of duplicates, topic distillation and classification, and interface design for presentation and navigation. Each problem is going to be addressed in the following sections.

3.1 Data Acquisition and Parsing

We are using a fixed number of data feeds from major portuguese mainstream media sources for news gathering. We opted to only include generic mainstream sources in this initial selection. This allowed us to avoid the major challenges faced in web crawling — link discovery and content encoding. Since the feeds are known in advance, we customized content decoding routines for each individual source. Also, since all sources publish web feeds in XML, the content extraction was straightforward. The fetching is scheduled to be performed periodically every hour on all sources. Thus, we need to account for news items that are downloaded multiple times. We use the URL (after handling redirects) of the news item to detect duplicate content. All content in stored in a UTF-8 encoded format on the server.

[1] `http://press.jrc.it/NewsExplorer`

[2] `http://labs.google.com/inquotes`

3.2 Quote Extraction

There is a variety of ways in which quotes can be expressed. Quotes can be found either in the title or throughout the body of the news feed. Some quotes are expressed *directly* using (supposedly) the speaker's exact words, while other are expressed *indirectly*, after being rephrased by the journalist. In some cases, the speaker is identified in a position that is adjacent to the quote (e.g. in the same sentence) while in other cases anaphoric resolution has to be made to find the correct speaker. This can involve, for example, matching an existing ergonym with the speaker's name (e.g. "The [Prime-Minister] said...."). Table 1 shows examples of some of the several different possible situations. In the current version of `verbatim` we mainly addressed quotes that explicitly mention the speaker in order to avoid anaphoric resolution. More specifically, we look for sentences in the body of the news feed that match the following pattern: *[Optional Ergonym], [Name of Speaker], [Speech Act] [Direct or Indirect Quote]* Using this pattern (and some minor variations of it), we are able to extract structures such as example (3) and (4) shown in Table 1. Because these structures are quite standard, the identification of each of the elements (ergonym, name of the speaker, speech act and quote) does not require extensive semantic analysis capabilities (such as noun-phrase identification and named-entity recognition), and can be achieved by using regular expressions and lists of words. Currently, extraction is made using 19 patterns (most small variations of the one previously shown), and a list with 35 speech acts (e.g. "afirmou", "acusou", "disse", etc.). In practice, about 5% of the news feeds match these patterns. Nevertheless, there

#	**Position**	**Direct?**	**Source**
1	title	yes	Costa Pinto: É óbvio que PS pedirá maioria absoluta
2	body	yes	"É indispensável uma ruptura com esta política de direita e é indispensável construir neste país uma alternativa à esquerda a estas políticas", afirmou Carlos Gonçalves, da Comissão Politica do PCP.
3	body	no	Hernâni Gonçalves também não acredita que os casos de corrupção que agora aparecem nas divisões distritais sejam a consequência do que se passa no futebol português ao mais alto nível.
4	body	mix	O vice-presidente do PSD, Aguiar-Branco, considerou que o primeiro-ministro perdeu uma oportunidade de "falar a verdade ao país".
5	body	yes	"Penso que um presidente ou presidente eleito e a sua equipa devem saber fazer várias coisas ao mesmo tempo. A propósito da situação em Gaza, sou colocado a par todos os dias", indicou aos jornalistas.

Table 1. Examples of several different types of quotes found in news objects.

are still many other quotes, with different structures, that we are not able to extract in the current version of `verbatim`.

3.3 Removal of Duplicates

Since `verbatim` processes news feeds from several sources, it is quite usual to extract duplicate or near duplicates news from which duplicate quotes will be extracted. Such duplicate quotes do not provide any additional information to the user. On the contrary, they pollute presentation of data and damage statistics (e.g.: who are the most active speakers?). Therefore, after extracting quotes from news feeds, we proceed by trying to aggregate the most similar quotes in *quote groups*, Q_1, Q_2,... Q_{last}, with one or more quotes. The longest quote in the group is considered the *head* of the group. Every time a new quote is found, q_{new} it is compared with the head quote of each of the k most recent quote groups: Q_{last}, Q_{last-1}, Q_{last-2}... $Q_{last-k+1}$. If the similarity between q_{new} and the head quote of any of such groups is higher that a given threshold, s_{min}, then q_{new} is added to the most similar group. Otherwise, a new group, Q_{new} is created, containing q_{new} only, which becomes the new head of the group (although possibly only temporarily). This procedure is a very simplified approach of the *streaming clustering* algorithm [1].

Comparison between the new quotes q_{new} and the head quote for group k, q^k_{head} is made in two steps. First, we check if the speaker of each quote is the same. Currently, we only check for exact lexical matches (which may be problematic when there are small lexical variations such as “Ehud Barak” and “Ehud Barack”). If the name of the speakers is the same, then similarity between quotes is measured by comparing the actual content of the quote. We obtain the vector representation of each quote using a binary *bag-of-words* approach (stop words are removed). Vectors are compared using the Jaccard Coefficient. When similarity is higher that $s_{min} = 0.25$, then quotes are considered near-duplicates. Table 2 shows some statistics about the sizes of 730 groups of quotes that result from the aggregation of a set of 1,359 quotes extracted from the news sources we are consulting (several topics are covered). Only 427 out of 1,359 quotes are found to be unique. For most of the cases, the number of duplicate and near-duplicate is considerable.

# Quotes in Group	# Groups	# Quotes in Group	# Groups
1	427	5	20
2	125	6	6
3	98	7	4
4	47	≥ 8	3

Table 2. Number of quotes in the groups for 1,359 extracted quotes.

3.4 Topic Classification

verbatim tries to a assign a *topic tag* to each quote. However, because there is a very wide variety of topics in the news, topic classification becomes a quite complex problem. In fact, the set of possible topics in news is open: unseen topics can be added as more news are collected. Thus, efficient topic classification of news requires that the system is able to (i) dynamically identify new topics tags as they appear in the news, (ii) automatically generate a training set using that includes examples for the new topic tags, and (iii) re-train the topic classification procedure accordingly.

Identification of Topic Tags and Generation of Training Set The identification of topic tags is made by mining a very common structure found in news titles: "*topic tag*: *title headline*". For example: "*Operação Furacão*: Acusações estão atrasadas, admite PGR", "*Música*: Morreu Ron Ashton, guitarrista dos Stooges", "*Telecom*: Acordo sobre Redes de Nova Geração assinado 4ª feira..." or "*Sócrates/Entrevista*: Primeiro-ministro esteve 'longe' da verdade...". From a set of about 26,000 news items, we were able to find 783 different topic tags (occuring in at least two titles). For illustration purposes, the top 20 most common topic tags up to the first week of January 2009 are presented in Table 3 (the distribution of topic tags changes with time). We are thus able to generate a training set that matches each topic tag t_i from the set of topic tags found, $\mathcal{T}$, with a set of news items for that topic, $I_i = (i_{1i}, i_{2i}... i_{Ni})$ (i.e those items that contained the topic tag in the title). We will denote the complete training set as the $\mathcal{T} \rightarrow \mathcal{I}$, mapping from $\mathcal{T} = (t_1, t_2... t_k)$ to $\mathcal{I} = (I_i, I_2,... I_k)$.

#	Topic Tag	# Titles Found	#	Topic Tag	# Titles Found
1	Médio Oriente	172	11	Guiné-Bissau	95
2	Futebol	164	12	BPN	90
3	Música	150	13	Tailândia	87
4	Crise	137	14	Casa Pia	83
5	Lisboa	133	15	Brasil	72
6	Índia	132	16	Manoel de Oliveira	69
7	EUA	124	17	Literatura	67
8	Educação	113	18	Espanha	62
9	Saúde	108	19	PSD	55
10	Cinema	104	20	Cultura	53

Table 3. Top 20 most common topic tags.

Training the Topic Classifiers We use two different text classification approaches to assign topic tags to quotes: *Rocchio classification* [7] and *Support*

Vector Machines (SVM) [2]. Both involve representing news feeds as vectors of features. We use a bag-of-words approach for vectorizing news feed items. Let i_k be a news item, composed of title string, i_k^{title}, and a body string i_k^{body}. Let $[(w^t_1, f^t_1), (w^t_2, f^t_2), ... (w^t_m, f^t_m)]$ and $[(w^b_1, f^b_1), (w^b_2, f^b_2), ... (w^b_n, f^b_n)]$ be the bag-of-words vector representation of i_k^{title} and i_k^{body} respectively. The vector representation of i_k is obtained by adding the bag-of-word descriptions of i_k^{title} and i_k^{body} while concatenating the prefix "t_" or "b_" to each of the features depending on whether they come from the title or the body:

$$[i_k] = [(t\_w^t_1, f^t_1), (t\_w^t_2, f^t_2), ...(t\_wt^t_m, f^t_m), (b\_w^b_1, f^b_1), (b\_w^b_2, f^b_2), ...(b\_w^b_n, f^b_n)] \tag{1}$$

This vector representation allows us to keep information about the source of each word since such information may be important in topic classification.

Rocchio classification provides a straight-forward way to classify items using a nearest-neighbour assignment. For each class c_i, of a set of $|C|$ classes, we obtain $[c_i]$, the vector representation of the class. $[c_i]$ is computed from the vector representation of items from that class, $[i_{ij}]$, taken from the training set. Usually, $[c_i]$ is the arithmetic average of vectors $[i_{ij}]$, i.e. the *vector centroid*, although other item aggregation and feature weighting schemes are possible. Having the vector representation for each of the $|C|$ classes, classification is made by comparing the vector representation of a new item, $[i^{new}]$, against all the vector descriptions of the $|C|$ classes, $[c_1]$, $[c_2]$ ... $[c_{|C|}]$. Usually, the cosine metric is used to compare $[i^{new}]$ with $[c_i]$, i.e. $cos([i^{new}], [c_i])$. Item $[i^{new}]$ is then classified as belonging to the class corresponding to the closest $[c_i]$ vector. We used a variation of TF-IDF weighting for producing class descriptions. For each topic tag t_j in the training set $\mathcal{T} \rightarrow \mathcal{I}$ we start by adding the vector representation (Equation 1) of the corresponding news items $i_{ij} \in I_j$:

$$[c^*_k] = \sum_i [i_{ik}] = \Big[(t\_w^t_1, F^t_1), ...(t\_wt^t_{|T|}, F^t_{|T|}), (b\_w^b_1, F^b_1), ...(b\_w^b_{|B|}, F^b_{|B|})\Big] \tag{2}$$

with F^t_1, F^b_i being the summation of individual frequencies found in each $[i_{ik}]$ for the $|T|$ and $|B|$ distinct word features extracted from the title and the body of the news items respectively. Let $f_{tpc}(w)$ be the *topic frequency* of word w, i.e. the number of vectors $[c^*_k]$ in which we find a non-nil component for the feature word w. Then, vector representation of class c_j, corresponding to topic tag t_j is given by:

$$[c_k] = [\left(t\_w^t_1, \frac{F^t_1}{f_{tpc}(t\_w^t_1)}\right), ... \left(t\_w^t_{|T|}, \frac{F^t_{|T|}}{f_{tpc}(t\_w^t_{|T|})}\right), \\ \left(b\_w^b_1, \frac{F^b_1}{f_{tpc}(b\_w^b_1)}\right), ... \left(b\_w^b_{|B|}, \frac{F^b_{|B|}}{f_{tpc}(b\_w^b_{|B|})}\right)] \tag{3}$$

Support Vector Machines (SVM) provide an alternative approach to classification. SVMs have proven to be quite effective in classification tasks where items are described by vectors in high-dimensional spaces, as is the case of text classification. A SVM efficiently finds the hyperplane that allows to separate items of two classes with *maximum* margin. SVMs are *binary classifiers*, and therefore require training multiple classifiers to perform classification under a multi-class scenario. Thus, for each topic tag t_k in the training set $\mathcal{T} \rightarrow \mathcal{I}$, we will need to train one SVM classifiers, using $I^{+}(k) = I_k$, the *positive examples*, and $I^{-}(k) = \mathcal{I} - I_k$, the *negative examples*:

$$svm_k = train_{svm}(I^{+}(k), I^{-}(k)) = train_{svm}(I_k, \mathcal{I} - I_k) \tag{4}$$

The training procedure converts each news item to its vector description $[i_k j]$ using the procedure explained before (Equation 1). After training for a given topic tag t_k, the corresponding svm_k, will produce a value from 1 to -1 when used to classify a given news item, i^{news}:

- $svm_k([i^{news}]) > 0$ if i^{news} belongs to class corresponding to topic t_k
- $svm_k([i^{news}]) < 0$ if i^{news} does not belong to class corresponding to topic t_k

We use the SVM-light [3] package to train an SVM for each topic, svm_k. Training is made using the package default parameters, and resulting SVM descriptions are saved in files for being used later, during the topic classification procedure.

Topic Classification Procedure The topic classification of a given quote is achieved by classifying all news item from which the quote was extracted. The intuition here is that using information from the entire news item should help us obtaining more evidence about which topic tag should be assigned to the quote. Thus, topic classification of a quote, is equivalent to topic classification of the news item i^{qt} from which the quote was extracted. Using the Rocchio classifier, classification is achieved by performing vector comparison between $[i^{qt}]$ (Equation 1) and the vector representation of all $|\mathcal{T}|$ classes, $[c_k]$ (Equation 3), and then choosing the tag that corresponds to the most similar $[c_k]$, as measured by the cosine metric, $cos([c_k], [i^{qt}])$. Likewise, classification using SVM's is made by feeding $[i^{qt}]$ to each of the $|\mathcal{T}|$ SVM classifiers, svm_k, and choosing the topic tag t_k that corresponds to $max(svm_k([i^{qt}]))$.

The two classifiers do not operate in the same way and, thus, do not always agree. Since we wish to obtain the most out of both classifiers, we developed a procedure for combining classification results. Let $\mathcal{T} = (t_1, t_2 ... t_k)$ be the set of topic tags over which the SVM and the Rocchio classifiers were trained, let i^{qt} be news items to classify, and let $[i^{qt}]$ be its vector representation. Then:

- compute svm_{max}, the maximum value given by $svm_k([i^{qt}])$, corresponding to $k = k^{svm}_{max}$.
- compute roc_{max} be the maximum value given by $cos([c_k], [i^{qt}])$, corresponding to $k = k^{roc}_{max}$.
- if $svm_{max} \geq min_{svm}$, then topic for i^{qt} will be $t_{k^{svm}_{max}}$
- else if $roc_{max} \geq min_{roc}$, then topic for i^{qt} will be $t_{k^{roc}_{max}}$
- else do not classify news item i^{qt}

This way, we give preference to classification made by the Support Vector Machines. We only refer to results provided by the Rocchio classifier when results from SVMs have low degree of confidence. In practice, we found that such combination procedure achieves reasonable results with $min_{svm} = -0.2$, and $min_{roc} = 0.2$. The negative value for min_{svm} arises from the fact that the number of positive examples used while training the SVMs, $\#(I^{+}(k))$, is always much less than the number of negative examples, $\#I^{-}(k) = \#(\mathcal{I} - I_k)$, leading to rather conservative classifiers, i.e. classifiers biased to produce false negatives. Still, for about 23% of the quotes we are not able to find any topic for which $svm_{max} \geq min_{svm}$, or $roc_{max} \geq min_{roc}$, These quotes remain unclassified and, therefore, are not shown to the user.

3.5 Web Interface

`verbatim`'s end-user web interface was developed over a standard LAMP technology stack. All data is stored on a MySQL database and published using CGI Perl scripts. Two interface screenshots are presented in figures 1 and 2. The homepage is shown in Figure 1 where several features are highlighted: (a) AJAX powered search over named entities (i.e. speakers); (b) last quotes found, ordered by number of supporting news; (c) most active topics, with the number of quotes in parentheses; (d) most active named entities, also with the number of quotes in parentheses.

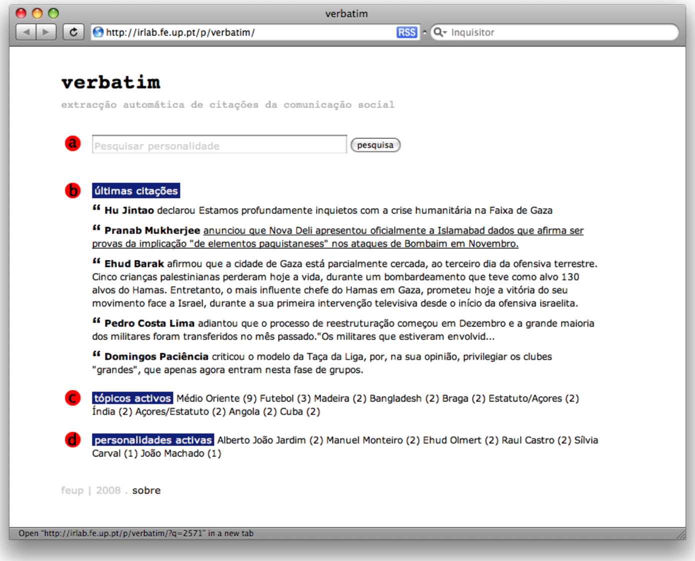
Each quote has an individual page where the news items used to support the quote are listed. Figure 2 shows the interface for the topic “Médio Oriente” (Middle East). For each topic, the following features are available for direct inspection: (a) last quotes extracted for this topic; (b) navigational links to filter by named entity within this topic; (c) named entity search box, available in all pages. There is also a similar page for each single named entity where additional filters by topic are available.

`verbatim` includes user-friendly URLs for search, topics and named entities. For instance, the see all of Barack Obama's quotes the URL is — `http://irlab.fe.up.pt/p/verbatim/?w=Barack+Obama`. Finally, we have also included a data access API based on Atom web feeds. We publish web feeds for the last extracted quotes, last quotes by a given named entity and last quotes for a given topic.

3.6 Overall Update Routine

Because news are constantly being produced, `verbatim` needs to be updated cyclically. There are two update routines: the *quote extraction* routine, which runs every hour, and the *classifier re-training* routine, which runs every 24 hours. The quote extraction routine is the following:

- Read web feeds made available from the chosen set of news providers and store parsed news items in local database;
- For each unprocessed item in the news database, run the quote extraction procedure. Store the extracted information (id of source news item, name



verbatim

extracção automática de citações da comunicação social

últimas citações

“ **Hu Jintao** declarou Estamos profundamente inquietos com a crise humanitária na Faixa de Gaza

“ **Pranab Mukherjee** anunciou que Nova Deli apresentou oficialmente a Islamabad dados que afirma ser provas da implicação "de elementos paquistaneses" nos ataques de Bombaim em Novembro.

“ **Ehud Barak** afirmou que a cidade de Gaza está parcialmente cercada, ao terceiro dia da ofensiva terrestre. Cinco crianças palestinianas perderam hoje a vida, durante um bombardeamento que teve como alvo 130 alvos do Hamas. Entretanto, o mais influente chefe do Hamas em Gaza, prometeu hoje a vitória do seu movimento face a Israel, durante a sua primeira intervenção televisiva desde o início da ofensiva israelita.

“ **Pedro Costa Lima** adiantou que o processo de reestruturação começou em Dezembro e a grande maioria dos militares foram transferidos no mês passado."Os militares que estiveram envolvid...

“ **Domingos Paciência** criticou o modelo da Taça da Liga, por, na sua opinião, privilegiar os clubes "grandes", que apenas agora entram nesta fase de grupos.

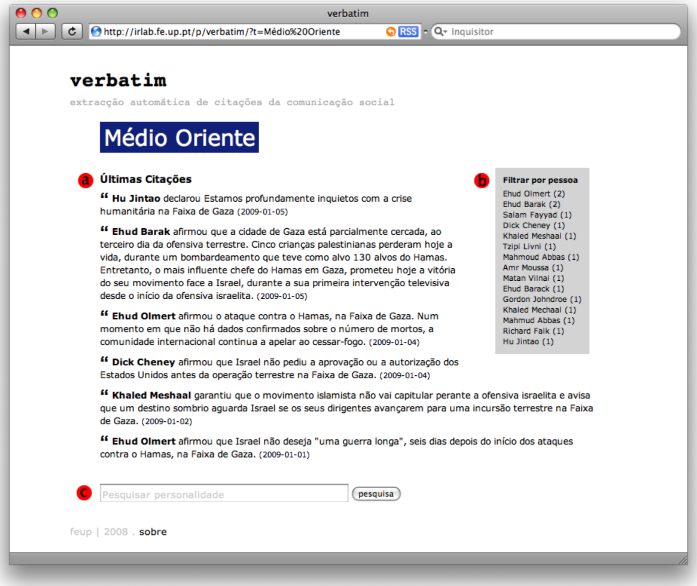
tópicos activos Médio Oriente (9) Futebol (3) Madeira (2) Bangladesh (2) Braga (2) Estatuto/Açores (2) Índia (2) Açores/Estatuto (2) Angola (2) Cuba (2)

personalidades activas Alberto João Jardim (2) Manuel Monteiro (2) Ehud Olmert (2) Raul Castro (2) Sílvia Carval (1) João Machado (1)

feup | 2008 . sobre

Open "http://irlab.fe.up.pt/p/verbatim/?q=2571" in a new tab

Fig. 1. `verbatim` homepage.



verbatim

extracção automática de citações da comunicação social

Médio Oriente

Últimas Citações

“ **Hu Jintao** declarou Estamos profundamente inquietos com a crise humanitária na Faixa de Gaza (2009-01-05)

“ **Ehud Barak** afirmou que a cidade de Gaza está parcialmente cercada, ao terceiro dia da ofensiva terrestre. Cinco crianças palestinianas perderam hoje a vida, durante um bombardeamento que teve como alvo 130 alvos do Hamas. Entretanto, o mais influente chefe do Hamas em Gaza, prometeu hoje a vitória do seu movimento face a Israel, durante a sua primeira intervenção televisiva desde o início da ofensiva israelita. (2009-01-05)

“ **Ehud Olmert** afirmou o ataque contra o Hamas, na Faixa de Gaza. Num momento em que não há dados confirmados sobre o número de mortos, a comunidade internacional continua a apelar ao cessar-fogo. (2009-01-04)

“ **Dick Cheney** afirmou que Israel não pediu a aprovação ou a autorização dos Estados Unidos antes da operação terrestre na Faixa de Gaza. (2009-01-04)

“ **Khaled Meshaal** garantiu que o movimento islamista não vai capitular perante a ofensiva israelita e avisa que um destino sombrio aguarda Israel se os seus dirigentes avançarem para uma incursão terrestre na Faixa de Gaza. (2009-01-02)

“ **Ehud Olmert** afirmou que Israel não deseja "uma guerra longa", seis dias depois do início dos ataques contra o Hamas, na Faixa de Gaza. (2009-01-01)

Filtrar por pessoa
Ehud Olmert (2)
Ehud Barak (2)
Salam Fayyad (1)
Dick Cheney (1)
Khaled Meshaal (1)
Tzipi Livni (1)
Mahmoud Abbas (1)
Amr Moussa (1)
Matan Vilnai (1)
Ehud Barack (1)
Gordon Johndroe (1)
Khaled Mechaal (1)
Mahmud Abbas (1)
Richard Falk (1)
Hu Jintao (1)

feup | 2008 . sobre

Fig. 2. `verbatim` example topic page.

of speaker, optional ergonym, speech act, direct or indirect quote) in the database;
- Run the quote duplicate detection routine to group duplicate quotes together;
- For each unclassified group of quotes (which can be a singleton), run the classification procedure. Store classification in the database. Unclassified quotes are kept in the database (but will not be visible to the user) since it might be possible to classify them when classifiers are re-trained.

The classifier re-training routine updates SVMs descriptions and Rocchio class vector descriptions. This step is required for including new topics that might be found in the most recent news items, and for refining the descriptions of already existing topics with more recent information. SVMs require a complete training from scratch (i.e. training is not incremental), which, given the number of topics (several hundreds) and the number of news items at stake (several thousands), may require significant CPU resources. Therefore, we opted for performing such retrain every 24 hours. The classifier update routine is the following:

- Take all news items from database and run the topic detection procedure in order to build the training set $\mathcal{T} \rightarrow \mathcal{I}$, for topics $\mathcal{T} = (t_1, t_2... t_k)$ and new items $\mathcal{I} = (I_i, I_2,... I_k)$;
- Vectorize all news items for all I_i in $\mathcal{I}$;
- Train Rocchio:
  - Compute topic frequency for all feature words;
  - For each t_k in $\mathcal{T}$ generate $[c*_k]$ by summing vectors associated with I_k, and obtain $[c_k]$ by weighing $[c*_k]$ with information about topic frequency for each feature word.
- Train SVM:
  - For each t_k in $\mathcal{T}$ generate $I^+(k) = I_k$, the set of *positive examples*, and $I^-(k) = \mathcal{I} - I_k$, the set of *negative examples* and make $svm_k = train_{svm}(I_k, \mathcal{I} - I_k)$.

The descriptions of Rocchio classes and SVMs are stored in file, and loaded when the topic classification routine is executed.

4 Results and Error Analysis

The database currently stores a total of 26,266 news items (as of early January 2009) after 47 days of regular fetching. During this period `verbatim` extracted a total of 570 quotes from 337 distinct named entities over 197 different topics. A ratio of roughly 1 distinct quote for every 46 news items. We found a small number of duplicate quotes being extracted as independent quotes (5 quotes). Figure 3 presents a plot of the number of news items together with the number of quotes over time.

We conducted a manual inspection over all quotes, named entities and topics to identify extraction errors. From a total of 570 quotes extracted, 68 were errors

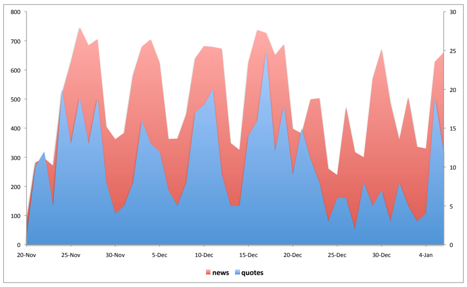


Fig. 3. News (left vertical axis) and extracted quotes (right) over time.

(11.9%). Also, in 337 named entities extracted, 6 (1.8%) were found to be errors. Finally, from the 197 topics identified, only 1 (0.5%) was an error. It is important to note that most of the errors found are minor and have little impact on the readability of the quote. Finally, to evaluate the algorithm for matching quotes to topics we also conducted a manual inspection over all quote/topic pairs. We found a total of 42 topics misattributed to quotes (7.4%).

5 Conclusion and Future Work

verbatim has proved to be a solid, fully functional online service working over live data from the portuguese mainstream media. Since the public launch of verbatim in mid November 2008, we have observed an average of 7.7 visits/day and a total of nearly 3,700 pageviews. The overall feedback, both online and offline, as been positive, and we plan to continue this line of research to improve verbatim in four main ways:

- **Increase the number of news sources**: after the initial proof of concept based on a small number of news feeds (8 in total), we plan to add a significant number of new sources to the database;
- **Improve quotations extraction**: as noted in the previous section, the ratio of quotations extracted is currently at 1 (distinct) quote for every 46 news items. Direct inspection of news items shows that this number can be easily improved by fine tuning additional matching rules, and creating new rules for other common pattern (both in news body and title). Using the information that we are able to gather about the mapping between names and ergonyms should also help increase the recall of extraction, since in many quotation there is no direct reference to the name but rather to the ergonym. Conflating variations of names for the same speaker, should also help to correctly group quotations of the same speaker;
- **Improving topic extraction and classification**: we found two problematic issues in topic extraction which require more attention. The first arises from the fact that news sources are not consistent about the words used to describe the topics in the headers (from which we extract the topic tag). This

leads extracting different topic tags from news, which in fact, refer to the same true topic (e.g. "Crise Finaceira", "Crise Económica"). In other cases, words used in news header refer to topics more generic than the true topic (e.g. "Desporto" instead of "Futebol"). Additionally, we would like to experiment using a different features to represent vectors, such as for example bigrams or part-of-speech tagged information.

- **Upgrade the end-user interface** As the number of quotes available in the database increase, the limitations of the current web interface become evident. For instance, as the archive of quotes by speaker becomes larger, the interface becomes cluttered. To overcome this problem we plan to add an additional navigational axis based on temporal information (e.g. filter by time feature).

Finally, we also plan to conduct a more extensive evaluation of `verbatim`'s performance by collecting a reference collection and computing traditional evaluation measures (i.e. Precision, Recall).

Acknowledgments Luís Sarmento and Sérgio Nunes were supported by Fundação para a Ciência e a Tecnologia (FCT) and Fundo Social Europeu (FSE - III Quadro Comunitário de Apoio), under grants SFRH/BD/23590/2005 and SFRH/BD/31043/2006.

References

1. Sudipto Guha, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering data streams. In *IEEE Symposium on Foundations of Computer Science*, pages 359–366, 2000.
2. Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.
3. Thorsten Joachims. Making large-scale svm learning practical. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999. software available at `http://svmlight.joachims.org/`.
4. Khoo Khyou and Bun Mitsuru Ishizuka. Topic extraction from news archive using tf*pdf algorithm. In *Proceedings of 3rd Int'l Conference on Web Informtion Systems Engineering (WISE 2002) ,IEEE Computer Soc*, pages 73–82. WISE, 2002.
5. Ralf Krestel, Sabine Bergler, and René Witte. Minding the source: Automatic tagging of reported speech in newspaper articles. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008)*. European Language Resources Association (ELRA), May 2008.
6. Bruno Pouliquen, Ralf Steinberger, and Clive Best. Automatic detection of quotations in multilingual news. In *Proceedings of Recent Advances in Natural Language Processing 2007*, Borovets, Bulgaria, 2007.
7. J. Rocchio. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System*, pages 313–323, Englewood, Cliffs, New Jersey, 1971. Prentice Hall.