JRC's Participation in the Guided Summarization Task at TAC 2010

Josef Steinberger, Hristo Tanev, Mijail Kabadjov, and Ralf Steinberger

Joint Research Centre, European Commission, Via E. Fermi 2749, Ispra (VA), Italy
firstname.lastname@jrc.ec.europa.eu

Abstract. In this paper we describe our participation in the Guided Summarization Task at the Text Analysis Conference 2010 (TAC'10). The goal of the task was to encourage a deeper semantic analysis of the source documents instead of relying only on document word frequencies to select important concepts. We used the output of our event extraction system and automatic learning of semantically-related terms to capture the required aspects of each particular article category. We submitted two runs: the first uses information extraction tools in combination with co-occurrence of features, the second uses only co-occurrence information. In the following sections we describe our runs and discuss the results attained.

1 Introduction

Our main goal is to produce multilingual summaries within the Europe Media Monitor (EMM)[1] framework. EMM gathers around 100,000 of news articles every day from various news sources. All news articles received are clustered at short regular intervals producing topic-homogeneous news clusters for each of the 40+ languages. Thus, the number of clusters can be big and the size of each cluster may vary from a dozen to more than a hundred articles. Multi-document summarization systems could potentially reduce this big bulk of highly redundant news data and obtain one succinct text which summarizes the most important content.

We participated in the previous TACs. In 2008, our lexical LSA-based approach (J. Steinberger and Ježek (2009)), which tries to capture the most important concepts in the source articles, was ranked 9th in overall responsiveness within the 58 participating systems. In 2009, we included named entities in the summarizer's input representation. It resulted in 2nd place among 52 runs (J. Steinberger et al. (2009a)). This TAC encourages an even deeper semantic analysis of the source documents by its new Guided summarization task. We were given a list of aspects for each article category, and the summary should include those aspects if possible.

The task naturally led to the integration with information extraction tools. We used our event extraction system that is focused on similar issues as the categories defined for TAC 2010. For capturing other aspects we automatically learned terms semantically-related to a manually created set of seed terms. The aim was to select frequently mentioned information, whilst at the same time making sure that this information also captures the required aspects. Thus, we combined the co-occurrence-based information from LSA with the aspect information coming from the event extraction system.

In section 2 we describe the information extraction tools we used. We continue with the description of the summarizer in section 3. In section 4 we present a detailed analysis of the results obtained in the TAC experiment and then we conclude.

[1] http://emm.jrc.it/overview.html

2 Information Extraction for Summarization

We describe here information extraction approaches we used to capture the required summary information. For capturing highly frequent topics in a cluster we use in addition to lexical features (words and bigrams) also person/organization/location entity mentions discovered by our entity recognition and disambiguation tools. For capturing the category-related aspects we used our event extraction system and the tool for automatic learning of semantic classes.

2.1 Entity Recognition and Disambiguation

Within the EMM's NewsExplorer project[2] R. Steinberger et al. (2009c) developed multilingual tools for geo-tagging Pouliquen et al. (2006) and entity disambiguation (Pouliquen and Steinberger, 2009). We used both systems to extract information about mentions of the entities in the TAC clusters. The extracted features were used as additional features in co-occurrence calculation but also to capture several aspects (places of events and persons involved in investigations).

2.2 Aspects Identified by NEXUS

NEXUS is an event extraction system which analyzes news articles reporting on violent events, natural or man-made disasters (see Tanev et al. (2008) for detailed system description). The system identifies the type of the event (e.g., flooding, explosion, assassination, kidnapping, air attack, etc.), number and description of the victims, as well as descriptions of the perpetrators and the means, used by them. For example for the text "Three people were shot dead and five were injured in a shootout", NEXUS will return an event structure with three slots filled: The *event type* slot will be set to *shooting*; the *dead victims* slot will be set to *three people*; and the *injured* slot will be set to *five*. Event extraction is deployed as a part of the EMM family of applications, described in Steinberger et al. (2009b).

NEXUS relies on a mixture of manually created linguistic rules, linear patterns, acquired through machine learning procedures, plus domain knowledge, represented as domain-specific heuristics and taxonomies. For example, one of the linear patterns for detection of *dead victims* is *[PERSON-GROUP] were shot dead* . The *[PERSON-GROUP]* phrases are recognized by a finite-state grammar. Event type detection is done through a combination of keywords, a Naive Bayes statistical classifier and several domain-specific rules.

NEXUS has been used to analyze online news in several languages and showed reasonable levels of accuracy Tanev et al. (2008).

We found out that some of the aspects, relevant to the summarization task, correspond to the information extracted by NEXUS. In particular, the aspects "What happened", "Perpetrators" and "Who affected" have corresponding slots in the event structures of NEXUS.

In our summarization experiments we ran the event extraction system on each news article from the corpus and we mapped extracted slots to summarization aspects. This was done in the following way: The event type (e.g., terrorist attack) was mapped to the aspect "What happened"; the slot "Perpetrator" was mapped to the aspect "Perpetrators"; and the values for the aspect "Victims" were obtained as a union of the event slots: "Dead victims", "Injured", "Arrested", "Displaced", "Kidnapped", "Released hostages" and "People, left without homes". At the end, from a fragment like: "three people died and many were injured", the system will extract two values for the aspect "Who affected", namely "three people" and "many".

[2] `http://emm.newsexplorer.eu/`

2.3 Learning Lexica for Aspect Recognition

Ontopopulis is a system for automatic learning of semantic classes (see Tanev et al. (2010) for algorithm overview and evaluation). As an input, it accepts a list of words, which belong to a certain semantic class, e.g. "disasters", then it learns additional words, which belong to the same class. Ontopopulis is a multilingual adaptation of a syntactic approach described earlier in Tanev and Magnini (2006). This approach accepts one or several seed sets of terms, each belonging to a semantic class; then, it finds other terms, which are likely to belong to the same semantic class.

Ontopopulis extracts for each semantic class a list of context features, n-grams which tend to occur with the seed set for this class. Each n-gram has a statistical score assigned to it. At the end, for each semantic class, the system finds other terms, which tend to co-occur with its context features. These terms are considered as candidate terms for the corresponding semantic class. For example, if we want to learn words from the class "natural disaster", we can give to Ontopopulis the following seed set *earthquake, flooding, tsunami*. Then, the system learns terms like *mudslides, landslide, tornado, cyclone, flash floods, fire, wildfires*, etc.

Clearly, the system output needs to be manually cleaned, in order to build an accurate lexicon. Since the terms are ordered by reliability (more reliable terms are on the top), the user can review the list, from top to down, deciding where to stop on the basis of his/her availability or the quality of the list around the point reached within the list. The unrevised items are discarded. Another possibility is to skip the manual reviewing process and take all the terms up to a certain threshold. This approach, however, cannot guarantee very high accuracy.

We learned 4 lexicons, using Ontopopulis, followed by manual cleaning. Each lexicon was relevant to a specific summary aspect. The four aspects covered by our lexicons are: "Damages", "Countermeasures", "Resource", and "Charges". Here we give a short sample from each of the learned lexicons:

1. Damages: damaged, destroyed, badly damaged, extensively damaged, gutted, torched, severely damaged, burnt, burned
2. Countermeasures: operation, rescue operation, rescue, evacuation, treatment, assistance, relief, military operation, police operation, security operation, aid
3. Resource: water, food, species, drinking water, electricity, gas, forests, fuel, natural gas
4. Charges: rape, kidnapping, aggravated, murder, attempted murder, robbery, aggravated assault, theft, armed robbery

The words and multi-words from these four lexicons were used to trigger the corresponding summary aspects.

3 Sentence Extraction Based on Co-occurrence and Aspect Information

In this section we describe how the extracted information is combined with lexical features to produce summaries that contain frequently mentioned information (derived from co-occurrence analysis) as well as the required aspects.

3.1 LSA-based Co-occurrence Information

Originally proposed by Gong and Liu (2002) and later improved by J. Steinberger and Ježek (2004), this approach first builds a term-by-sentence matrix from the source, then applies

Singular Value Decomposition (SVD) and finally uses the resulting matrices to identify and extract the most salient sentences.

The LSA approach to summarization first builds a term-by-sentence matrix from the source, then applies Singular Value Decomposition (SVD) and finally uses the resulting matrices to identify and extract the most salient sentences. SVD finds the latent (orthogonal) dimensions, which in simple terms correspond to the different topics discussed in the source.

More formally, we first build matrix $\mathbf{A}$ where each column represents the weighted term-frequency vector of sentence j in a given set of documents. The weighting scheme we found to work best is using a binary local weight and an entropy-based global weight (for details see Steinberger and Ježek (2009)). If we generalize the notion of term to entail, in addition to words, also entities we can obtain a semantically enriched representation.

After that step Singular Value Decomposition (SVD) is applied to the above matrix as $\mathbf{A} = \mathbf{USV}^T$, and subsequently matrix $\mathbf{F} = \mathbf{S} \cdot \mathbf{V}^T$ reduced to r dimensions[3] is derived. This matrix that is passed to the sentence selection phase represents the topics of the cluster identified by co-occurring features.

3.2 Aspect Information

We use the aspects identified by the information extraction tools to boost the co-occurrence-based scores of the sentences that contain the aspects relevant to the corresponding cluster category. For each article cluster we build aspect-by-sentence matrix $\mathbf{P}$ which contains boolean values to store aspects' presence/absence in sentences. For each cluster category a different set of aspects is applied. This matrix is used in the sentence selection process then.

3.3 Sentence Selection

Input to the sentence scoring/selection is formed by matrices $\mathbf{F}$, containing information about the most important topics within the cluster, and $\mathbf{P}$, containing aspect information.

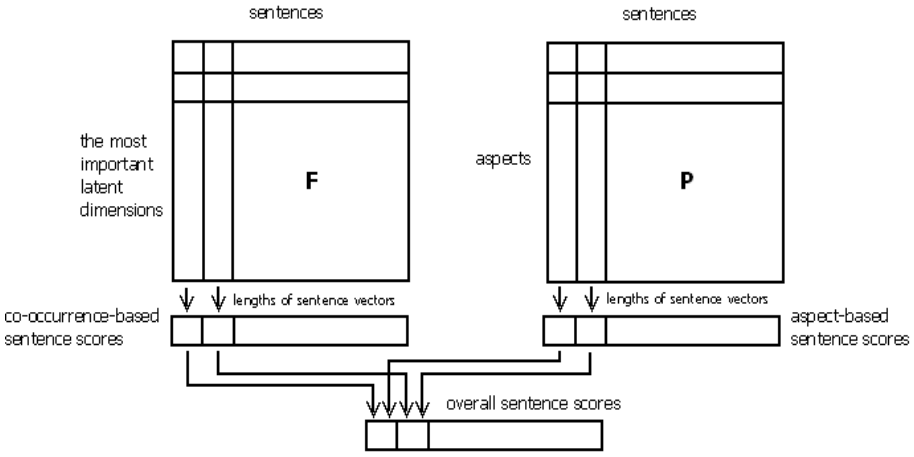

Fig. 1. Sentence selection process.

[3] The degree of importance of each 'latent' topic is given by the singular values and the optimal number of latent topics (i.e., dimensions) r can be fine-tuned on training data.

Sentence selection (see figure 1) starts with measuring the length of sentence vectors in matrix $\mathbf{F}$. The length of the vector can be viewed as a measure for importance of that sentence within the top cluster topics. We call it 'co-occurrence sentence score'. For the aspect matrix ($\mathbf{P}$) we do the same: measuring the length of sentence vectors. In this case the score corresponds to how many relevant aspects the sentences contain ('aspect-based sentence score'). The two scores are then combined in a way that the aspect-based score works as a booster for the co-occurrence score. The formula for the overall score computation is defined as follows:

$$o_j = |\mathbf{f}_j|(1 + |\mathbf{a}_j|^{bc}). \tag{1}$$

where o_j is the overall score of sentence j, $|\mathbf{f}_j|$ and $|\mathbf{a}_j|$ are its corresponding vectors lengths in matrices $\mathbf{F}$ and $\mathbf{P}$. Coefficient bc can control the impact of aspects on the overall score.

The sentence with the largest overall score is selected as the first to go in the summary (its corresponding vector in $\mathbf{F}$ is denoted as $\mathbf{f}_{best}$, similarly $\mathbf{p}_{best}$ for $\mathbf{P}$). After placing it in the summary, the topic/sentence distribution in matrix $\mathbf{F}$ is changed by subtracting the information contained in that sentence:

$$\mathbf{F}^{(it+1)} = \mathbf{F}^{(it)} - \frac{\mathbf{f}_{best} \cdot \mathbf{f}_{best}^T}{|\mathbf{f}_{best}|^2} \cdot \mathbf{F}^{(it)}, \tag{2}$$

The vector lengths of similar sentences are decreased, thus preventing within summary redundancy. For aspects, however, we wish to select diverse information as well. But we take a different approach for that. There are cases in which the same aspect should be repeated. For example, for a killing event we want to see the date of the killing and the date when the perpetrator was arrested. Another example are countermeasures. Both following snippets were found important in a model summary of TAC'09 data: *Russian rescue attempts to free and raise the submarine were unsuccessful. Russia requested international help.* Thus, we lower the influence of the aspects already contained in the summary but we do not zero it. Also, we do not use the same formula as in the case of matrix $\mathbf{F}$ because here we are in positive low-dimensional space in comparison with the positive/negative high-dimensional LSA latent space. We use the following formula to update each value in matrix $\mathbf{P}$:

$$\mathbf{p}_{i,j}^{(it+1)} = dc * \mathbf{p}_{i,j}^{(it)}, \quad \text{if} \quad \mathbf{p}_{i,best}^{(it)} > 0. \tag{3}$$

By dc we can control the fadeout of used aspects (a value from 0 to 1).

After the subtraction of information in the selected sentence the process continues with the sentence which has the largest overall score computed from updated matrices $\mathbf{F}$ and $\mathbf{P}$. The process is iteratively repeated until the required summary length is reached.

3.4 The Case of Update Summarization

We used the same approach to produce update summaries. However, in this case when creating the aspect matrix we considered only those aspect items that do not occur in the basic documents. For instance, if we find in the basic document set that "200 people were killed", this string is not considered as an update item if found in the update document set. However, if there is more specific information like "212 people were killed" it is considered as the update aspect item.

4 Results and Discussion

The task was similar to the last year's one: to write a 100-word summary for a set of 10 newswire articles for a given topic, where the topic falls into a predefined categories. However, participants (and human summarizers) were given a list of important aspects for each category, and a summary had to cover all those aspects if possible. The summaries could have also contained other information relevant to the topic.

There was an update part of the task this year as at TAC'08 and TAC'09: to write a 100-word update summary of a subsequent 10 newswire articles for the topic, under the assumption that the user has already read the earlier articles.

The defined categories and its aspects were the following:

1. Accidents and Natural Disasters:
   - 1.1 WHAT: what happened
   - 1.2 WHEN: date, time, other temporal placement markers
   - 1.3 WHERE: physical location
   - 1.4 WHY: reasons for accident/disaster
   - 1.5 WHO AFFECTED: casualties (death, injury), or individuals otherwise negatively affected by the accident/disaster
   - 1.6 DAMAGES: damages caused by the accident/disaster
   - 1.7 COUNTERMEASURES: countermeasures, rescue efforts, prevention efforts, other reactions to the accident/disaster
2. Attacks (Criminal/Terrorist):
   - 2.1 WHAT: what happened
   - 2.2 WHEN: date, time, other temporal placement markers
   - 2.3 WHERE: physical location
   - 2.4 PERPETRATORS: individuals or groups responsible for the attack
   - 2.5 WHY: reasons for the attack
   - 2.6 WHO AFFECTED: casualties (death, injury), or individuals otherwise negatively affected by the attack
   - 2.7 DAMAGES: damages caused by the attack
   - 2.8 COUNTERMEASURES: countermeasures, rescue efforts, prevention efforts, other reactions to the attack (e.g., police investigations)
3. Health and Safety:
   - 3.1 WHAT: what is the issue
   - 3.2 WHO AFFECTED: who is affected by the health/safety issue
   - 3.3 HOW: how they are affected
   - 3.4 WHY: why the health/safety issue occurs
   - 3.5 COUNTERMEASURES: countermeasures, prevention efforts
4. Endangered Resources:
   - 4.1 WHAT: description of resource
   - 4.2 IMPORTANCE: importance of resource
   - 4.3 THREATS: threats to the resource
   - 4.4 COUNTERMEASURES: countermeasures, prevention efforts
5. Investigations and Trials (Criminal/Legal/Other):
   - 5.1 WHO: who is a defendant or under investigation
   - 5.2 WHO INV: who is investigating, prosecuting, or judging
   - 5.3 WHY: general reasons for the investigation/trial
   - 5.4 CHARGES: specific charges to the defendant
   - 5.5 PLEAD: defendant's reaction to charges, including admission of guilt, denial of charges, or explanations

5.6 SENTENCE: sentence or other consequences to defendant

We used several types of information extraction for capturing the aspects. Several aspects were identified by our event extraction system:

- WHAT HAPPENED (used for aspects 1.1, 2.1, 3.1, 5.3): = type of event (e.g. 'bombing');
- WHO AFFECTED (1.5, 2.6, 3.2, 5.1) = number of victims/injured/ displaced etc. (we extracted a full string, not only a number, e.g. '200 soldiers killed');
- PERPETRATORS (2.4, 5.1).

We treated the aspect 5.1 in a special way. For several event types, like 'arrest' the affected person is the one who is investigated, however, for other types of events like 'killing' that person is the perpetrator. This is the reason why we used both WHO AFFECTED and PERPETRATORS slots for capturing the aspect.

The lexical lists of semantically similar terms were generated for capturing the following aspects:

- DAMAGES (1.6, 2.7);
- COUNTERMEASURES (1.7, 2.8, 3.5, 4.4);
- RESOURCE (4.1) = list of resources;
- CHARGES (5.4).

For the identification of temporal expressions (aspects 1.2, 2.2) we produced simple lists of month names etc. Now we work on including a proper temporal analysis.

In the case of aspect 5.2 we took advantage of the fact that we have information about person mentions in the text. This aspect was set in the case that there was a person mentioned in the particular sentence. We took the same approach for locations (1.3 an 2.3). All locations were considered as fillers of that aspect.

We did not deal with the most complex aspects (1.4, 2.5, 3.3, 3.4, 4.2, 4.3, 5.5, 5.6). We simply rely on the fact that they should be captured by the co-occurrence part of the sentence scorer if they seem to be important (frequently mentioned).

We submitted two runs. The first one (assigned by no. 25) is the complete proposed system: combines co-occurrence and aspect information. The second run (no. 31) represents our baseline system: uses only co-occurrence information (including lexical and entity co-occurence). In the remainder of this discussion we refer to the former run as the IE run and the latter as non-IE run (but note that the non-IE run includes the named entity information).

The summaries were evaluated at NIST for content (based on Columbia University's Pyramid method (Nenkova and Passonneau, 2004)), readability/fluency and overall responsiveness. ROUGE (Lin, 2004) and BE (Hovy et al., 2005) scores were also provided.

The total number of systems this year was 43 including two baselines. The 1st baseline (LEAD) was the first 100 words from the most recent document, the 2nd baseline was the output of the MEAD summarizer (Radev et al., 2003). 23 groups participated.

We can analyze 3 types of results. The overall results compare the systems based on all 46 topics (clusters) - basic and update summaries. We have also results for each category. But also, we can see how well we identified each aspect (only pyramid scores are available).

4.1 Overall Results

Tables 1 and 2 contain the overall TAC results for initial and update summaries. We report the results and corresponding ranks (in brackets) within all the 43 systems of two best TAC systems, our two submissions, and the two baselines.

In the case of initial summaries the run that included aspects (run 25) performed better in the overall responsiveness and linguistic quality than the run based on co-occurrence only (run

Run ID	Overall responsiveness	Linguistic quality	Pyramid score
16 (the best run in Overall resp.)	**3.17** (1)	3.46 (2)	0.40 (4)
22 (the best run in Pyramid score)	3.13 (2)	3.11 (13)	**0.43** (1)
25 (co-occurrence+aspects)	2.98 (10)	3.35 (4)	0.37 (18)
31 (co-occurrence only)	2.89 (19)	3.28 (6)	0.38 (13)
2 (baseline - MEAD)	2.50 (27)	2.72 (29)	0.30 (26)
1 (baseline - LEAD)	2.17 (32)	**3.65** (1)	0.23 (32)

Table 1. TAC'10 results of the Guided summarization task - initial summaries.

31). It was slightly worse when evaluated by the Pyramid method. We do not report here the evaluation of the number of repetitions, but also in this qualitative measure the aspect-based run was better. The reason could be that we try to select diverse aspects here. Overall, both our systems were ranked high in linguistic quality. One reason could be that sentences that contain full entity mentions, which are used as features in the co-occurrence-based part of the sentence scorer, are getting higher scores. They are usually summary-worthy sentences and do not contain links (e.g., pronouns) to entities in the preceding context. Our systems performed better than both baselines, with the obvious exception of the LEAD baseline and linguistic quality (the summary is formed by a continuous text from one article). The score differences between our systems and the best two listed systems (16 and 22) were not significant.

Run ID	Overall responsiveness	Linguistic quality	Pyramid score
16 (the best run in Overall resp.)	**2.72** (1)	3.33 (2)	**0.32** (1)
13 (the best run in Pyramid score)	2.65 (2)	3.24 (6)	0.32 (2)
25 (co-occurrence+aspects)	2.28 (19)	3.09 (11)	0.23 (20)
31 (co-occurrence only)	2.35 (15)	3.30 (3)	0.25 (12)
2 (baseline - MEAD)	2.48 (6)	2.70 (30)	0.26 (10)
1 (baseline - LEAD)	2.02 (28)	**3.74** (1)	0.19 (28)

Table 2. TAC'10 results of the Guided summarization task - update summaries.

By looking at the results of update summaries we can observe that the run that does not use information extraction to capture the aspects performed better. However the differences are not substantial. The run was also ranked high in the linguistic quality. However, the results seem to be more negative because there is a larger gap compared to the best TAC systems and even the MEAD baseline performed slightly better than our systems with the exception of linguistic quality. The LEAD baseline was worse than most of the systems.

In what follows we will focus only on the discussion of the basic summaries' results. Though, similar figures could be also created for update summaries.

4.2 Category-focused Results

Now we continue with the discussion of the results for each category. We report the scores and ranks of both our systems in each cell of the table – the first score and rank correspond to Run 25 (with information extraction-based aspect capturing), the second to Run 31 (co-occurrence only).

Category	Overall responsiveness	Linguistic quality	Pyramid score
1. Disasters	3.00 (23) - **3.57 (2)**	3.43 (3) - 3.29 (5)	0.38 (23) - **0.43 (10)**
2. Attacks	**3.71 (3)** - 2.86 (22)	**3.29 (4)** - 3.00 (16)	**0.56 (6)** - 0.49 (18)
3. Health	**2.75 (6)** - 2.42 (21)	3.33 (6) - 3.25 (9)	0.30 (9) - 0.31 (7)
4. Resources	2.50 (25) - 2.60 (21)	3.60 (3) - 3.40 (6)	0.24 (29) - 0.27 (23)
5. Investigations	3.20 (6) - 3.30 (2)	3.10 (10) - **3.40 (2)**	0.45 (14) - 0.47 (5)

Table 3. Scores and ranks of our runs for each category (run 25 – run 31).

In the case of category "Accidents and natural disasters" the co-occurrence-only approach worked clearly better than the approach with IE. Our simpler run was ranked 2nd in overall responsiveness. The reason of the weaker performance of the IE-based run could be that several times the summarizer selected a sentence that mentioned a historical event, not the event that the cluster was focused on (like a previous earthquake in the same place).

On the contrary, in attacks we can see a really huge improvement with IE: 6th in Pyramids (compared to 18th), 3rd in overall resp. (compared to 22nd) and 4th in linguistic quality (compared to 16th). It could be explained by the fact that this category is the focus of the event extraction system.

In the 'health and safety' category we can notice an improvement when using IE, except of Pyramids. Overall, the runs were ranked high in that category. In the case of 'endangered Resources' the results were poor. We did not focus on this particular category. The linguistic quality, however, showed high levels also for this category.

In the last category, investigations and trials, the system without IE worked better but the differences in the scores were not significant. Our simpler system was ranked high: 2nd in both linguistic quality and overall responsiveness, and 5th in Pyramids.

4.3 Aspect-focused Results

In this section we focus on the most fine-grained results: how well each particular aspect was captured. We can use only Pyramid scores for this evaluation. We report the scores and ranks of our systems and the score of the best system. However, the best score refers to a different system for each aspect.

Firstly, we look at the aspects derived from NEXUS (table 4). Clearly, using type of event as capturing the 'what happened' aspect was not successful. An indicator like 'bombing' seem to be too general for capturing what happened. This aspect could be left on LSA to select the most frequent information. In the case of aspect 'who affected' there was a large improvement for the attacks category. Roughly speaking, there was no effect in other categories. We noticed also an improvement in update summaries for this aspect. The IE run was successful in capturing also the 'perpetrators' aspect in comparison with the run without IE. Compared to other systems, however, the runs were ranked only slightly above the average.

Aspect	Run 25 (rank)	Run 31 (rank)	Best
1.1 WHAT (disasters)	0.60 (24)	**0.79 (3)**	0.89
2.1 WHAT (attacks)	0.74 (21)	0.79 (12)	0.88
3.1 WHAT (health)	0.33 (17)	0.36 (14)	0.58
5.3 REASONS (investigations)	0.46 (19)	**0.59 (6)**	0.67
1.5 WHO AFFECTED (disasters)	0.36 (25)	0.41 (23)	0.68
2.6 WHO AFFECTED (attacks)	**0.65 (2)**	0.54 (11)	0.66
3.2 WHO AFFECTED (health)	0.29 (6)	0.31 (4)	0.39
5.1 WHO (investigations)	0.67 (17)	0.65 (19)	0.96
2.4 PERPETRATORS (attacks)	**0.48 (18)**	0.34 (24)	0.69

Table 4. Pyramid scores and ranks of our runs for each aspect identified by the event extraction system.

Next, we look at the aspects derived from the lexical list generated by Ontopopulis (table 5). In the case of damages we can see worse results with IE in the disasters' category. Treating all events in the cluster as equal probably led to selecting sentences, and subsequently also damages, concerned with non-central events. In attacks we can observe, that without IE we did not capture any damage (the score is 0), compared to the 4th best performance with IE. 'Countermeasures' was the category where the IE-based run was very successful in all four categories. It suggests the lexical lists were the right choice for treating this aspect. In resource descriptions there was an non-significant improvement with IE. In capturing charges the co-occurrence information itself performed better.

Aspect	Run 25 (rank)	Run 31 (rank)	Best
1.6 DAMAGES (disasters)	0.13 (26)	**0.38 (10)**	1.25
2.7 DAMAGES (attacks)	**0.50 (4)**	0 (30)	0.75
1.7 COUNTERMEASURES (disasters)	**0.34 (7)**	0.19 (29)	0.39
2.8 COUNTERMEASURES (attacks)	**0.34 (18)**	0.20 (32)	0.65
3.5 COUNTERMEASURES (health)	**0.31 (1)**	0.24 (7)	0.31
4.4 COUNTERMEASURES (resources)	**0.36 (5)**	0.29 (12)	0.50
4.1 WHAT (resources)	0.49 (19)	0.46 (25)	0.81
5.4 CHARGES (investigations)	0.33 (27)	**0.47 (11)**	0.72

Table 5. Pyramid scores and ranks of our runs for each aspect identified by generated lexical lists.

Among the aspects which were treated by other ways (table 6) the only successful was the 'who involved' aspect in investigations. Actually, giving a larger weight to all person mentions made a great job, ranking our IE-based submission as the best one. Treating the place aspect the same way was not successful. For capturing time of the events the co-occurrence-driven approach worked well in the case of attacks (2nd).

There are several complex aspects on which we have not worked yet. However, in table 7 we can find how well it can be caught by the co-occurrence analysis. In several aspects we received surprisingly good results (the best in reasons for attacks) but for example in the 'importance of resource' aspect we did not capture anything.

Aspect	Run 25 (rank)	Run 31 (rank)	Best
1.2 WHEN (disasters)	0.25 (21)	**0.35 (13)**	0.60
2.2 WHEN (attacks)	0.48 (9)	**0.62 (2)**	0.67
1.3 WHERE (disasters)	0.39 (26)	**0.55 (15)**	0.78
2.3 WHERE (attacks)	0.70 (14)	0.67 (16)	0.86
5.2 WHO INV (investigations)	**0.55 (1)**	0.39 (5)	0.55

Table 6. Pyramid scores and ranks of our runs for all other treated aspects.

Aspect	Run 25 (rank)	Run 31 (rank)	Best
1.4 WHY (disasters)	**0.40 (4)**	0.32 (12)	0.50
2.5 WHY (attacks)	**0.69 (1)**	0.61 (2)	0.69
3.4 WHY (health)	0.11 (26)	0.16 (23)	0.55
3.3 HOW (health)	0.23 (24)	0.26 (19)	0.49
4.2 IMPORTANCE (resources)	0 (39)	0 (39)	0.45
4.3 THREATS (resources)	0.17 (33)	0.19 (30)	0.38
5.5 PLEAD (investigations)	**0.41 (4)**	0.35 (9)	0.48
5.6 SENTENCE (investigations)	0.14 (22)	0.17 (20)	0.51

Table 7. Pyramid scores and ranks of our runs for all other (not treated) aspects.

5 Conclusion

We used the event extraction tool to feed the co-occurrence-based summarizer with semantic information about category-focused aspects. The results showed its great impact on the clusters that deal with the central focus of the event extraction system - criminal/terrorist attack. In natural disasters the approach has to firstly classify the extracted aspect information as topic-central or background (historic) and then to feed the central items to the summarizer. We noticed good results in most of the aspects treated by the automatically generated lexical lists. This suggests using this approach even for other semantic classes where we can identify a reasonable base of seed terms. A proper temporal analysis can improve the time-related aspect and we believe it can strongly improve the update information identification and the discovery of the central event of the topic. The results however also showed that there is still a huge gap between the human and automatic summaries in all evaluation metrics.

Bibliography

Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of ACM SIGIR*, New Orleans, US, 2002.

E. Hovy, C. Lin, and L. Zhou. Evaluating duc 2005 using basic elements. In *Proceedings of the DUC*, 2005.

C.-Y. Lin. ROUGE: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, Barcelona, Spain, 2004.

A. Nenkova and R. Passonneau. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2004.

B. Pouliquen and R. Steinberger. Automatic construction of multilingual name dictionaries. In Cyril Goutte, Nicola Cancedda, Marc Dymetman, and George Foster, editors, *Learning Machine Translation*. MIT Press, NIPS series, 2009.

B. Pouliquen, M. Kimler, R. Steinberger, C. Ignat, T. Oellinger, K. Blackler, F. Fuart, W. Zaghouani, A. Widiger, A. Forslund, and C. Best. Geocoding multilingual texts: Recognition, disambiguation and visualisation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 53–58, Genoa, Italy, May 2006.

D. Radev, J. Otterbacher, H. Qi, and D. Tam. Mead reducs: Michigan at duc 2003. In *Proceedings of DUC 2003*, 2003.

J. Steinberger and K. Ježek. Text summarization and singular value decomposition. In *Proceedings of the 3rd ADVIS conference*, Izmir, Turkey, 2004.

J. Steinberger and K. Ježek. Update summarization based on novel topic distribution. In *Proceedings of the 9th ACM Symposium on Document Engineering, Munich, Germany*, 2009.

J. Steinberger, M. Kabadjov, B. Pouliquen, R. Steinberger, and M. Poesio. WB-JRC-UTs participation in tac 2009: Update summarization and aesop tasks. In *Proceedings of the Text Analysis Conference (TAC)*, 2009a.

R. Steinberger, B. Pouliquen, and Erik Van der Goot. An introduction to the europe media monitor family of applications. In *Information Access in a Multilingual World Proceedings of the SIGIR*, 2009b.

R. Steinberger, B. Pouliquen, and C. Ignat. Using language-independent rules to achieve high multilinguality in text mining. In François Fogelman-Soulié, Domenico Perrotta, Jakub Piskorski, and Ralf Steinberger, editors, *Mining Massive Data Sets for Security*. IOS-Press, Amsterdam, Holland, 2009c.

H. Tanev and B. Magnini. Weakly supervised approaches for ontology population. In *Proceedings of the 11th conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2006.

H. Tanev, J. Piskorski, and M. Atkinson. Real-time news event extraction for global crisis monitoring. In *Proceedings of 13th International Conference on Applications of Natural Language to Information Systems (NLDB 2008)*, 2008.

H. Tanev, V. Zavarella, J. Linge, M. Kabadjov, J. Piskorski, M. Atkinson, and R.Steinberger. Exploiting machine learning techniques to build an event extraction system for portuguese and spanish. *Journal Linguamatica: Revista para o Processamento Automatico das Linguas Ibericas*, 2010.