Using Mechanical Turk to Create a Corpus of Arabic Summaries

Mahmoud El-Haj, Udo Kruschwitz, Chris Fox

School of Computer Science and Electronic Engineering
University of Essex
Colchester, CO4 3SQ
United Kingdom
{melhaj,udo,foxcj}@essex.ac.uk

Abstract

This paper describes the creation of a human-generated corpus of extractive Arabic summaries of a selection of Wikipedia and Arabic newspaper articles using Mechanical Turk—an online workforce. The purpose of this exercise was two-fold. First, it addresses a shortage of relevant data for Arabic natural language processing. Second, it demonstrates the application of Mechanical Turk to the problem of creating natural language resources. The paper also reports on a number of evaluations we have performed to compare the collected summaries against results obtained from a variety of automatic summarisation systems.

1. Motivation

The volume of information available on the Web is increasing rapidly. The need for systems that can automatically summarise documents is becoming ever more desirable. For this reason, text summarisation has quickly grown into a major research area as illustrated by the Text Analysis Conference (TAC) and the Document Understanding Conference (DUC) series.

We are interested in the automatic summarisation of Arabic documents. Research in Arabic is receiving growing attention but it has widely been acknowledged that apart from a few notable exceptions—such as the Arabic Penn Treebank[1] and the Prague Arabic Dependency Treebank[2]—there are few publicly available tools and resources for Arabic NLP, such as Arabic corpora, lexicons and machine-readable dictionaries, resources that are common in other languages (Diab et al., 2007) although this has started to change in recent years (Maegaard et al., 2008; Alghamdi et al., 2009). Some reasons for this lack of resources may be due to the complex morphology, the absence of diacritics (vowels) in written text and the fact that Arabic does not use capitalisation. Tools and resources however are essential to advance research in Arabic NLP. In the case of summarisation tasks, most of the activity is concerned with the English language—as with TAC and DUC. This focus is reflected in the availability of resources: in particular, there is no readily available "gold standard" for evaluating Arabic summarisers.

Tools and resources are essential to advance research in Arabic NLP, but generating them with traditional techniques is both costly and time-consuming. It is for this reason that we considered using Amazon's Mechanical Turk[3]—an online marketplace for work that requires human intelligence—to generate our own reference standard for extractive summaries.

2. Related Work

There are various approaches to text summarisation, some of which have been around for more than 50 years (Luhn, 1958). These approaches include single-document and multi-document summarisation. One of the techniques of single-document summarisation is summarisation through extraction. This relies on the idea of extracting what appear to be the most important or significant units of information from a document and then combining these units to generate a summary. The extracted units differ from one system to another. Most of the systems use sentences as units while others work with larger units such as paragraphs.

Evaluating the quality and consistency of a generated summary has proven to be a difficult problem (Fiszman et al., 2009). This is mainly because there is no obvious ideal summary. The use of various models for system evaluation may help in solving this problem. Automatic evaluation metrics such as ROUGE (Lin, 2004) and BLEU (Papineni et al., 2001) have been shown to correlate well with human evaluations for content match in text summarisation and machine translation (Liu and Liu, 2008; Hobson et al., 2007, for example). Other commonly used evaluations include measuring information by testing readers' understanding of automatically generated summaries.

This very brief review of related work should serve as a motivation for the corpus of Arabic summaries that we have produced for the Arabic NLP community. Our decision to use the Mechanical Turk platform is justified by the fact that it has already been shown to be effective for a variety of NLP tasks achieving expert quality (Snow et al., 2008; Callison-Burch, 2009, for example).

3. The Document Collection

The document collection used in the development of the resource was extracted from the Arabic language version of Wikipedia[4] and two Arabic newspapers; Alrai[5] from Jordan and Alwatan[6] from Saudi Arabia. These sources were chosen for the following reasons.

[1] http://www.ircs.upenn.edu/arabic/

[2] http://ufal.mff.cuni.cz/padt/PADT_1.0/

[3] http://www.mturk.com

[4] http://www.wikipedia.com

[5] http://www.alrai.com

[6] http://www.alwatan.com.sa

1. They contain real text as would be written and used by native speakers of Arabic.

2. They are written by many authors from different backgrounds.

3. They cover a range of topics from different subject areas (such as politics, economics, and sports), each with a credible amount of data.

The Wikipedia documents were selected by asking a group of students to search the Wikipedia website for arbitrary topics of their choice within given subject areas. The subject areas were: art and music; the environment; politics; sports; health; finance and insurance; science and technology; tourism; religion; and education. To obtain a more uniform distribution of articles across topics, the collection was then supplemented with newspaper articles that were retrieved from a bespoke information retrieval system using the same queries as were used for selecting the Wikipedia articles. Each document contains on average 380 words.

4. The Human-Generated Summaries

The corpus of extractive document summaries was generated using Mechanical Turk. The documents were published as “Human Intelligence Tasks” (HITS). The assessors (workers) were asked to read and summarise a given article (one article per task) by selecting what they considered to be the most significant sentences that should make up the extractive summary. They were required to select no more than half of the sentences in the article. Using this method, five summaries were created for each article in the collection. Each of the summaries for a given article were generated by different workers.

In order to verify that the workers were properly engaged with the articles, and provide a measure of quality assurance, each worker was asked to provide up to three keywords as an indicator that they read the article and did not select random sentences. In some cases where a worker appeared to select random sentences, the summary is still considered as part of the corpus to avoid the risk of subjective bias.

The primary output of this project is this corpus of 765 human-generated summaries that we obtained, which is now available to the community.[7] To set the results in context, and illustrate its use, we also conducted a number of evaluations.

5. Evaluations

To illustrate the use of the human-generated summaries from Mechanical Turk in the evaluation of automatic summarisation, we created extractive summaries of the same set of documents using a number of systems, namely:

Sakhr: an online Arabic summariser.[8]

AQBTSS: a query-based document summariser based on the vector space model that takes an Arabic document and a query (in this case the document’s title) and returns an extractive summary (El-Haj and Hammo, 2008; El-Haj et al., 2009).

Gen-Summ: similar to *AQBTSS* except that the query is replaced by the document’s first sentence.

LSA-Summ: similar to *Gen-Summ*, but where the vector space is tranformed and reduced by applying Latent Semantic Analysis (LSA) to both document and query (Dumais et al., 1988).

Baseline-1: the first sentence of a document.

The justification for selecting the first sentence in *Baseline-1* is the believe that in Wikipedia and news articles the first sentence tends to contain information about the content of the entire article, and is often included in extractive summaries generated by more sophisticated approaches (Baxendale, 1958; Yeh et al., 2008; Fattah and Ren, 2008; Katragadda et al., 2009).

When using Mechanical Turk on other NLP tasks, it has been shown that aggregation of multiple independent annotations from non-experts can approximate expert judgement (Snow et al., 2008; Callison-Burch, 2009; Albakour et al., 2010, for example). For this reason, we evaluated the results of the systems not with the raw results of Mechanical Turk, but with derived *gold standard* summaries, generated by further processing and analysis of the human generated summaries.

The aggregation of the summaries can be done in a number of ways. To obtain a better understanding of the impact of the aggregation method on the results of the evaluation, we constructed three different gold standard summaries for each document. First of all we selected all those sentences identified by at least three of the five annotators (we call this *Level 3* summary). We also created a similar summary which includes all sentences that have been identified by at least two annotators (called *Level 2*). Finally, each document has a third summary that contains all sentences identified by any of the annotators for this document (called *All*). This last kind of summary will typically contain outlier sentences. For this reason, only the first two kinds of aggregated summaries (*Level 2* and *Level 3*) should really be viewed as providing genuine gold standards. The third one (*All*) is considered here just for the purposes of providing a comparison.

A variety of evaluation methods have been developed for summarisation systems. As we are concerned with *extractive* summaries, we will concentrate on results obtained from applying Dice’s coefficient (Manning and Schütze, 1999), although we will discuss briefly results from N-gram and substring-based methods ROUGE (Lin, 2004) and AutoSummENG (Giannakopoulos et al., 2008).

5.1. Dice’s Coefficient

We used Dice’s coefficient to judge the similarity of the sentence selections in the gold-standard extractive summaries — derived from the human-generated, Mechanical Turk summaries — with those generated by *Sakhr*, *AQBTSS*, *Gen-Summ*, *LSA-Summ* and *Baseline-1* (Table 1). Statistically significant differences can be observed in a number

[7] `http://privatewww.essex.ac.uk/~melhaj/easc.htm`

[8] `http://www.sakhr.com`

	Sakhr	AQBTSS	Gen-Summ	LSA-Summ	Baseline-1
All	39.07%	32.80%	39.51%	39.23%	25.34%
Level 2	48.49%	39.90%	48.95%	50.09%	26.84%
Level 3	43.40%	38.86%	43.39%	42.67%	40.86%

Table 1: Dice results: systems versus MTurk-derived gold standards.

	Sakhr	AQBTSS	LSA-Summ	Gen-Summ	Baseline-1
Sakhr	—	51.09%	58.77%	58.82%	38.11%
AQBTSS	51.09%	—	54.61%	58.48%	47.86%
LSA-Summ	58.77%	54.61%	—	84.70%	34.66%
Gen-Summ	58.82%	58.48%	84.70%	—	34.99%

Table 2: Dice results: comparing systems.

of cases, but we will concentrate on some more general observations.

We observe that the commercial system *Sakhr* as well as the systems that build a summary around the first sentence most closely approximate the gold standards, i.e. *Level 2* and *Level 3*. This is perhaps not surprising as the overlap with the document's first sentence has been shown to be a significant feature in many summarisers (Yeh et al., 2008; Fattah and Ren, 2008).

It is interesting to note that summaries consisting of a single sentence only (i.e. *Baseline-1*) do not score particularly well. That suggests that the first sentence is important but not sufficient for a good summary. When comparing *Baseline-1* with the *Level 2* and *Level 3* summaries, respectively, we also note how the "wisdom of the crowd" seems to converge on the first sentence as a core part of the summary.

Finally, the system that most closely approximates our *Level 2* gold standard uses LSA, a method shown to work effectively in various NLP and IR tasks including summarisation, e.g. (Steinberger and Ježek, 2004; Gong and Liu, 2001).

We also compared the baseline systems with each other (Table 2). This is to get an idea of how closely the summaries each of these systems produce correlate with each other. The results suggest that the system that extracts the first sentence only does not correlate well with any of the other systems. At the same time we observe that *Gen-Summ* and *LSA-Summ* generate summaries that are highly correlated. This explains the close similarity when comparing each of these systems against the gold standards (see Table 1). It also demonstrates (not surprisingly) that the difference between a standard vector space approach and LSA is not great for the relatively short documents in a collection of limited size.

5.2. Other Evaluation Methods

In addition to using Dice's coefficient, we also applied the ROUGE (Lin, 2004) and AutoSummENG (Giannakopoulos et al., 2008) evaluation methods.

In our experiments with AutoSummENG we obtained values for "CharGraphValue" in the range 0.516–0.586. This indicates how much the graph representation of a model summary overlaps with a given peer summary, taking into account how many times two N-grams are found to be neighbours. *Gen-Summ* and *LSA-Summ* gave the highest values indicating that they produce results more similar to our gold standard summaries than what Sakhr and *AQBTSS* produced.

When applying ROUGE we considered the results of ROUGE-2, ROUGE-L, ROUGE-W, and ROUGE-S which have been shown to work well in single document summarisation tasks (Lin, 2004). In line with the results discussed above, *LSA-Summ* and *Gen-Summ* performed better on average than the other systems in terms of recall, precision and F-measure (when using *Level 2* and *Level 3* summaries as our gold standards). Regarding the other systems, they all performed better than *Baseline-1*.

These results should only be taken to be indicative. Dice's coefficient appears to be a better method for *extractive* summaries as we are comparing summaries on the *sentence* level. It is however worth noting that the main results obtained from Dice's coefficient are in line with results from ROUGE and AutoSummENG.

6. Conclusions and Future Work

We have demonstrated how gold-standard summaries can be extracted using the "wisdom of the crowd".

Using Mechanical Turk has allowed us to produce a resource for evaluating Arabic extractive summarisation techniques at relatively low cost. This resource is now available to the community. It will provide a useful benchmark for those developing Arabic summarisation tools. The aim of the work described here was to create a relatively small but usable resource. We provided some comparison with alternative summarisation systems for Arabic. We have deliberately made no attempt in judging the individual quality of each system. How this resource will be used and how effective it can be applied remains the task of the users of this corpus.

7. References

M-D. Albakour, U. Kruschwitz, and S. Lucas. 2010. Sentence-level attachment prediction. In *Proceedings of the 1st Information Retrieval Facility Conference*, Lecture Notes in Computer Science 6107, Vienna. Springer.

M. Alghamdi, M. Chafic, and M. Mohamed. 2009. Arabic language resources and tools for speech and natural language: Kacst and balamand. In *2nd International Conference on Arabic Language Resources & Tools*, Cairo, Egypt.

P. B. Baxendale. 1958. Machine-made index for technical literature—an experiment. *IBM Journal of Research and Development*, 2.

C. Callison-Burch. 2009. Fast, cheap, and creative: Evaluating translation quality using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 286–295. Association for Computational Linguistics.

M. Diab, K. Hacioglu, and D. Jurafsky. 2007. Automatic Processing of Modern Standard Arabic Text. In A. Soudi, A. van den Bosch, and G. Neumann, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*, Text, Speech and Language Technology, pages 159–179. Springer Netherlands.

S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman. 1988. Using latent semantic analysis to improve access to textual information. In *CHI '88: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–285. ACM.

M. El-Haj and B. Hammo. 2008. Evaluation of query-based Arabic text summarization system. In *Proceeding of the IEEE International Conference on Natural Language Processing and Knowledge Engineering, NLP-KE'08*, pages 1–7, Beijing, China. IEEE Computer Society.

M. El-Haj, U. Kruschwitz, and C. Fox. 2009. Experimenting with Automatic Text Summarization for Arabic. In *Proceedings of the 4^{th} Language and Technology Conference (LTC'09)*, pages 365–369, Poznań, Poland.

M.A. Fattah and Fuji Ren. 2008. Automatic text summarization. In *Proceedings of World Academy of Science*, volume 27, pages 192–195. World Academy of Science.

M. Fiszman, D. Demner-Fushman, H. Kilicoglu, and T. C. Rindflesch. 2009. Automatic summarization of medline citations for evidence-based medical treatment: A topic-oriented evaluation. *Jouranl of Biomedical Informatics*, 42(5):801–813.

G. Giannakopoulos, V. Karkaletsis, G. Vouros, and P. Stamatopoulos. 2008. Summarization system evaluation revisited: N-gram graphs. *ACM Transactions on Speech and Language Processing (TSLP)*, 5(3):1–39.

Y. Gong and X. Liu. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 19–25. ACM.

S. P. Hobson, B. J. Dorr, C. Monz, and R. Schwartz. 2007. Task-based evaluation of text summarization using relevance prediction. *Information Processing & Management*, 43(6):1482–1499.

R. Katragadda, P. Pingali, and V. Varma. 2009. Sentence position revisited: a robust light-weight update summarization 'baseline' algorithm. In *CLIAWS3 '09: Proceedings of the Third International Workshop on Cross Lingual Information Access*, pages 46–52, Morristown, NJ, USA. ACL.

C. Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 25–26.

F. Liu and Y. Liu. 2008. Correlation between rouge and human evaluation of extractive meeting summaries. In *HLT '08: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, pages 201–204. ACL.

H. P. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research Development*, 2(2):159–165.

B. Maegaard, M. Atiyya, K. Choukri, S. Krauwer, C. Mokbel, and M. Yaseen. 2008. Medar: Collaboration between european and mediterranean arabic partners to support the development of language technology for arabic. In *In Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

C. D. Manning and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2001. BLEU: a method for automatic evaluation of machine translation. In *Proceeding of the 40th Annual Meeting on Association for Computational Linguistics (ACL'02)*. Association for Computational Linguistics.

R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. 2008. Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics.

J. Steinberger and K. Ježek. 2004. Using latent semantic analysis in text summarization and summary evaluation. In *Proceedings of the 5th International Conference on Information Systems Implementation and Modelling (ISIM)*, pages 93–100.

J.-Y. Yeh, H.-R. Ke, and W.-P. Yang. 2008. iSpreadRank: Ranking sentences for extraction-based summarization using feature weight propagation in the sentence similarity network. *Expert Systems with Applications*, 35(3):1451 – 1462.