Case-Based Approach to Automated Natural Language Generation for Obituaries

Ashish Upadhyay[1(✉)], Stewart Massie[1], and Sean Clogher[2]

[1] Robert Gordon University, Aberdeen, UK
{a.upadhyay,s.massie}@rgu.ac.uk
[2] The Obituary Company, Aberdeen, UK
theobituarycompany.contact@gmail.com

Abstract. Automated generation of human readable text from structured information is challenging because grammatical rules are complex making good quality outputs difficult to achieve. Textual Case-Based Reasoning provides one approach in which the text from previously solved examples with similar inputs is reused as a template solution to generate text for the current problem. Natural Language Generation also poses a challenge when evaluating the quality of the text generated due to the high cost of human labelling and the variety in potential good quality solutions. In this paper, we propose two case-based approaches for reusing text to automatically generate an obituary from a set of input attribute-value pairs. The case-base is acquired by crawling and then tagging existing solutions published on the web to create cases as problem-solution pairs. We evaluate the quality of the text generation system with a novel unsupervised case alignment metric using normalised discounted cumulative gain which is compared to a supervised approach and human evaluation. Initial results show that our proposed evaluation measure is effective and correlates well with average attribute error evaluation which is a crude surrogate to human feedback. The system is being deployed in a real-world application with a startup company in Aberdeen to produce automated obituaries.

Keywords: Natural Language Generation · Textual Case-Based Reasoning · Text evaluation

1 Introduction

Text generation from structured information is a common requirement for problem solving in variety of tasks and domains, such as compiling incident reports, writing customer reviews, and presenting weather forecasts [4,9,12]. These use-case examples typically have a common problem representation in that the generated text is the combination of the structured data (a set of pre-defined attribute values) and textual content, required to improve human readability. In this paper

we address a similar task in which a text generation system is required to automatically generate an obituary based on information about the deceased's life. The information typically includes: personal details; relationships, such as next-to-kin, spouse, children, friends; and details about funeral arrangements for the funeral or memorial.

The effectiveness of text generation system depend on the quality of the text produced, in terms of accuracy and readability, as well as the diversity of texts generated from the system. One approach is to use a standard abstract template with all the pre-defined attributes available as slots to be filled. But having a single template for every problem are difficult to construct for complex scenarios and result in very repetitive text outputs. Textual Case Based Reasoning (TCBR) gives an opportunity to develop dynamic templates with diverse text by re-using previous experiences.

In general, a TCBR system has a case-base containing information about previous experiences as its central knowledge source, which is used together with other key knowledge sources: the case representation and similarity knowledge [14]. In combination these knowledge sources enable the retrieval of similar cases from the case-base, providing a mechanism to re-use knowledge captured in previous examples to solve a new problem. Thus TCBR, as with CBR more generally, relies on the basic principle that "similar problems have similar solutions" [1]. Supervised Machine Learning approaches take advantage of this principle to learn more tailored representation or retrieval knowledge in order to improve some evaluation metric e.g. accuracy. However, in TCBR learning from labelled solutions is difficult because each solution tends to be unique and so simple feedback metrics are not so easily available to either refine or evaluate developing systems. We introduce a novel approach to evaluation that measures the extent to which similar problems have similar solutions by investigating the alignment between local neighbourhoods in the problem and solution space. This approach reduces the requirement for human evaluations.

In this work we generate a case-base by crawling the web to extract obituaries from Funeral Notices websites[1]. The information extracted from the website is plain text and needs pre-processing for building the case-base. In particular generating a structured representation in a knowledge rich manner. By manually analysing the processed obituaries, relevant attributes are identified to provide alternative representations for the problem component of the cases. An unsupervised evaluation technique is developed to evaluate the alternatives.

The main contributions of the work are as follows:

1. developing a real world system based on a TCBR approach for automatically generating obituaries which is being deployed by a start-up company;
2. a novel technique for evaluation of text generation with TCBR employing a case alignment approach using normalised discounted cumulative gain; and
3. demonstrating the effectiveness of the approach with experiments and comparison of results with other baselines and an average attribute error as a crude surrogate to human feedback.

---

[1] https://funeral-notices.co.uk/.

The rest of the paper is organised as follows. The problem domain is discussed in more detail in Sect. 2 before relevant related works are highlighted in the Sect. 3. The proposed case-based methodology for generation of textual obituaries identifies our alternative approaches to representation and similarity measuring in Sect. 4. The experimental design is discussed in Sect. 5, where we also introduce our novel evaluation method. In the Sect. 6, we discuss the results obtained from our experiments, before concluding the paper and looking at future works in Sect. 7.

2 Obituary Generation

An obituary is a written announcement of someone's death which is traditionally published in a local newspaper to inform the wider community about the death. It generally outlines the life and personality of the deceased person and provides the details of the funeral arrangements and memorials. In the growing digital era, people are tending towards using digital website to publish the obituaries instead of local newspapers to expand the audience from a local community to the wider world on the internet.

There are approximately 57,000 deaths in Scotland each year, of these two sites are providing obituaries notices currently. The main site captures only 10% of all death notices. There is an opportunity to improve the service provided and to integrate the latest AI technologies to support Funeral Directors to help the next of kin with the creation of digital public obituary notices.

Our commercial partner is in the process of providing a publication platform for obituary generation that focuses on supporting a sympathetic acknowledgement of the recently departed as a digitisation of the traditional print obituaries. In this paper, we investigated utilising a TCBR approach to generate dynamic and individual obituaries that help the next of kin prepare their tribute. The aim is to achieve a two-minute publication timeline, through an intuitive form that will lead to the generation of five bespoke obituary options, the undertaker and family can select the appropriate option with the ability to edit as required. New solutions generated on the system can be retained to increase the case-base size and diversity of solutions available.

A large number (around 100k) of obituaries, dating back to the year 2000, have been crawled and extracted from the web. As initial pre-processing, 30k obituaries created after 2015 are selected and out of these, the top 1000 notices based on those with higher word count is selected. After analysing this data an obituary can be divided into at least three distinct components: the personal information component; the relationships component; and funeral component.

1. **Personal Information**: this component gives the personal details of the deceased person, e.g. name, age, date and place of death, and cause of death. It can also include the information about the person's home town or previous working places, as well their hobbies.

2. **Relations**: this component presents the relatives' details, e.g., spouse, children, grand children, or in-laws. This component may also contain an emotional message about the family & friends and how the person is going to be missed by all who knew them.
3. **Funeral**: this component provides the details of funeral arrangements and will typically have the date, time and place for the memorial service. The component will also provide the information about the delivery of flowers and the potential guest list. For example, flowers may only be welcome from family members but all the friends and relatives are welcome to join at the memorial service. Options for donations and charity name can also be provided in this section in the lieu of flowers.

The main task for this project is to generate five diverse textual messages (obituary) based on the features given by the user. A simple message can be generated using an abstract template but then there will be no diversity in the generations and all the obituaries will become monotonous. The challenge is to generate human readable natural text which includes (almost) every feature to the generation and is diverse in nature as well.

3   Related Works

Automated generation of human readable text from structured data has been studied in various domains [6,9,13]. The studies mainly focus on the difficulties of mapping unstructured text from previous experience to a structured representation, measuring semantic or synaptic similarity for the retrieval & reuse of previous cases and automated evaluation of the generated text.

3.1   Text Generation

In [2], the author proposed a CBR system to generate weather forecast texts using examples from previous cases with similar weather states. For the retrieval of similar cases it is necessary to have same number of weather states in the retrieved one and the input query. The system fails to return a result if there's any mismatch in the number of states in input query and previous similar data. The system uses NIST5 score for evaluation requiring substantial reference texts for better performance. In [5] the textual summary of time series were generated using an end-to-end CBR system. The summary generation involved two steps where first an abstraction of time series is generated which in turns help the system to generate the textual summary of that abstraction. The system generated text was evaluated using a modified version of the edit distance measure [10] which heavily relies on the domain specifications. This is a custom evaluation approach that is difficult to use across different TCBR domains.

3.2   Case Alignment

There have been several approaches to measuring the performance of unsupervised CBR systems which focus on measuring the extent to which the problem-side space and solution-side space of case representation align with each other. In [7], authors proposed a case cohesion alignment to evaluate the performance of a CBR system which measures the level of overlap in retrieval set. However, the method requires a trial and error approach to set up a threshold for selecting the number of nearest neighbours in both the sets. A mechanism of case alignment was presented in [9] where the alignment was measured by taking the average solution similarity of its neighbours weighted by their problem-side similarities. Authors in [17] modified the case alignment measure by utilising the case ranking of similar cases in problem and solution sets by using a modified version of Kendall tau distance. Although the method works well in several CBR problems, it fails to scale in a TCBR scenario [17].

In this work, previous examples are marked up to act as dynamic templates which can be populated with structured data to generate good quality, diverse natural text. Alternative representations and similarity measures are compared. We also evaluate the quality of the generated text with a problem-solution alignment measure but propose a novel, domain independent metric taken from information retrieval.

4   Case-Based Methodology

Central to developing a CBR system is the availability of experiential knowledge that can provide previously solved successful examples for reusing to solve new problems. The crawled examples from the web provide a suitable source of past examples. However as obituaries in natural language they provide a case solution example but not with separate problem and solution representations required for CBR systems. The first task in developing a TCBR system is to create a case representation to effectively capture case knowledge as associated problem and solution components. The second stage is to develop a similarity metric utilising the problem representation to support retrieval.

4.1   Case Representation

In TCBR, cases are generally represented in two parts: problem and solution component. The problem representation comprises a set of attributes whose values can either be extracted from the crawled obituaries or are known for a new problem. The solution representation is the natural language text of the obituary but may be considered as a template with the associated problem attribute values identifed and replaced by mark-up tags.

For example given an obituary: "*OLIVIA WILSON, Peacefully on the 14th May 2019 at home, Olivia of Patna. Beloved wife to the late James Wilson, much loved mum to Jack and partner Emily, gran to Ava, Lucy and Logan, loving aunt,*

sister and a friend to all. Funeral service will be help at Patna Kirk, Patna on Monday 26th May, 2019 at 11.00am and thereafter to Patna Cemetery to which all friends are respectfully invited. Donations if desired to Cancer Research UK and Strathcarron Hospice.", Fig. 1 shows the case representation marked-up with attribute-value pairs in XML format.

```xml
<obit>
    <personal_info_component>
        <name>OLIVIA WILSON</name>,
        <demise_how>Peacefully</demise_how> on the
        <demise_date>14th May 2019</demise_date> at
        <demise_place>home</demise_place>,
        <nick_name>Olivia</nick_name> of
        <home_town>Patna</home_town>.
    </personal_info_component>
    <relations_component>
        Beloved
        <spouse_gender>wife</spouse_gender> to
        <spouse_name>the late James Wilson</spouse_name>, much loved
        <parent_gender>mum</parent_gender> to
        <children_name>Jack</children_name> and partner
        <children_in_law_name>Emily</children_in_law_name>,
        <grandparent_gender>gran</grandparent_gender> to
        <grandchildren_name>Ava, Lucy and Logan</grandchildren_name>, loving
        <other_relations_types>aunt</other_relations_types>,
        <siblings_gender>sister</siblings_gender> and a friend to all.
    </relations_component>
    <funerla_component>
        Funeral service will be help at
        <funeral_place>Patna Kirk, Patna</funeral_place> on
        <funeral_date>Monday 26th June, 2017</funeral_date> at
        <funeral_time>11.00am</funeral_time> and thereafter to
        <cemetery_place>Patna Cemetery</cemetery_place> to which
        <guests_list>all friends</guests_list> are respectfully invited. Donations if desired to
        <charity_name>Cancer Research UK and Strathcarron Hospice</charity_name>.
    </funerla_component>
</obit>
```

Fig. 1. Representation of a case marked-up in XML format.

Hence, an obituary contains information, as attribute values, on the different people, relationships, places, organisations, etc involved, and can be used to build an effective case representation that will be helpful for identifying similar cases to new problems. Around 40 relevant attributes have been selected to represent an obituary as a case in the case-base, as shown in Fig. 2 [2]. From the example obituary, we can see that the first sentence talks about the personal details of the deceased person, followed by relatives in second sentence and funeral information in the last sentence. This is a typical paragraph construction, so we can divide all the extracted obituaries into three components and annotate them with the identified attributes.

The attributes identified for annotations are set to be gender independent. For example, in Fig. 1 we have taken "*mum*" as a value for attribute "*parent_gender*". That means, the deceased person was parent (in this case mother)

---

[2] For the columns marked M/O: Mandatory/Optional, '-': Attribute value filled automatically based on the deceased's gender.

to "*Jack*" ("*children_name*"). So if we have a target problem with "*parent_gender* → *father*", the case in fig. 1 can still be re-used as a possible solution. An initial case-base has been created to seed the system by manually annotating 100 samples.

Attribute Name	Identifier Tag	Attribute Type	M/O	Comment
Name	< name >	String	M	
Gender	< gender >	Binary	M	Male (1) or Female (2)
Age	< age >	Number	O	Numbers in range 110
Demise Date	< demise_date >	Date	M	
Demise Place	< demise_place >	String	O	
Demise How	< demise_how >	3 Category	O	Peacefully (1), Suddenly (2), Peacefully but suddenly (3)
Demise Reason	< demise_reason >	Dropdown List	O	Name of any specific illness or accident
Home Town	< home_town >	String	O	Deceased Person Home Town
Nick Name	< nick_name >	String	O	
Occupation	< occupation >	String	O	Most recent job
Previous Works	< previous_works >	String	O	any previous achievements/-works

(a) Personal Information

Attribute Name	Identifier Tag	Att. Type	M/O	Comment
Spouse Name	< spouse_name >	String	O	
Spouse Gender	< spouse_gender >	Binary	-	Husband or Wife
Children Name	< children_name >	Array	O	
Parent Gender	< parent_gender >	String	-	Father or Mother
Grandchildren Name	< grandchildren_name >	Array	O	
Grandparent gender	< grandparent_gender >	String	-	Grandpa or Grandmother
Siblings Name	< siblings_name >	Array	O	
Siblings gender	< siblings_gender >	String	-	Brother or sister
Children-in-law Name	< children_in_law_name >	Array	O	
Parent-in-law gender	< parent_in_law_gender >	String	-	Father or Mother in law
Siblings-in-law Name	< siblings_in_law_name >	Array	O	
Siblings-in-law gender	< siblings_in_law_gender >	String	-	Brother or sister in law
Other Relations	< other_relations_names >	Array	O	
Friends Name	< friends_name >	Array	O	Name of the friends

(b) Relations Details

Attribute Name	Identifier Tag	Attribute Type	M/O	Comment
Funeral Place	< funeral_place >	String	M	
Funeral Date	< funeral_date >	Date	M	
Funeral Time	< funeral_time >	Time	M	
Cemetery Place	< cemetery_place >	String	O	
Cemetery Time	< cemetery_time >	Time	O	
Funeral Flowers	< flowers >	Binary	M	Family flowers only or welcome from all
Guests List	< guests_list >	Category	M	Public or private
Attire Request	< funeral_attire >	String	O	Any specific kind of attire
Donation	< charity_name >	String	O	Charity name for any donation
Associated Message	< funeral_message >	String	O	If the user wants to drop some message

(c) Funeral Details

Fig. 2. Attributes used for representation of obituaries.

4.2   Similarity Measure for Retrieval

We investigate two variants of similarity measure for retrieval of similar cases for a target problem. The first approach is straight-forward, where we match the number of features in the target problem with number of features in each case from the case-base. The first similarity measure ($sim1$) is defined by Eq. (1).

$$sim1 = |q \cap c| \qquad (1)$$

where q is the list of attributes in target problem and c is the list of attributes in each case from case-base.

There can be a problem with Eq. (1) where the target case has fewer features than the case retrieved from the case-base. Let's take an example where the target case has only 10 attributes out of a possible 40. In that scenario, cases with more than the 10 attributes will also have the same similarity score as cases

with the exact 10 features. To counter this problem, we use a different similarity measure ($sim2$), which is the Jaccard Similarity Coefficient (J) described in Eq. (2).

$$sim2 = J(q,c) = \frac{|q \cap c|}{|q \cup c|} \qquad (2)$$

4.3   Text Reuse

In the previous section we observed that there can be problems in situations where there is a misalignment in the number of attribute values. We used a different similarity measure to address this problem in Eq. (2). However, this method can also lead to a problem. The set of retrieved cases for a target problem with very less attributes might have same number of attributes but have different attribute types. For example, for a target problem with only "*spouse name*" in "*relation section*" and "*funeral place/time*" in "*funeral section*" along with all the attributes from "*personal info section*", the retrieved cases might contain only "*name*" and "*home town*" of the deceased person along with all the attributes from "*funeral section*". In that way the number of attributes may be the same giving a high similarity score for the retrieved case but in practice, it is not a good example of a similar case for re-use.

To address this problem, we investigate an alternative case representation where the case-base is broken down into three components, namely: *personal info component*; *relations component*; and *funeral component*. In this way, we can leverage our data and to find good retrieval examples with fewer cases. These components can also be broken down further into different sub-components such as: *relations component* could be separated in *spouse component*, *parent component*, and *grandparent component*. But this further breaking will reduce the attribute count in each sub-component and hence resulting into non-diverse case retrieval for every target problem, which will lead into generation of similar kind of text from the system every time. Thus, we need to find a balanced number of components for breaking the obituary representations.

With these insights, we propose two kinds of case retrievals for text reuse:

– **Basic**: retrieving whole obituary as an one entity; and
– **Component**: retrieving cases as 3 different components.

In Fig. 1, we can see that the three components are marked-up with separate "*component tags*". For *basic retrieval*, whole obituary is retrieved as one entity thus ignoring the component tags while for *component retrieval*, all three components are retrieved separately. Then the retrieved case's text is reused by replacing the each attribute's value with the attribute's value from the target problem. The new modified text is the solution generated by CBR system. In case of component retrieval, we combine the texts generated for each component separately to propose the final solution.

4.4 Solution Adaptation

The proposed solution may contain some general mistakes such as: referring to the deceased person with male pronoun even if the gender of the person is female or vice-versa; or adding an attribute which is not given in the target problem. These kinds of error occur because of the fact that the proposed solution is generated only by simply reusing the text from solution-side of the retrieved case after replacing the attributes' values.

To tackle the gender problem, we apply a rule-based adaptation process where each generated text is checked against the gender of the deceased person. If a pronoun with different gender is found in the text, it is replaced with the same pronoun of the deceased person's gender. For the extra attributes problem, we simply replace the attribute's value with a blank for any attribute which is not given in the target problem.

5 Experimental Evaluation

Evaluation of our TCBR system is a challenging task. It is difficult to automatically measure the effectiveness of a system due to the diversity found in the natural language output. Human evaluation is an alternative which, while effective is expensive and very time consuming. Traditional machine translation and summarising metrics such as BLEU [11] and ROUGE [8] scores are unlikely to work well because these metrics are based on the overlap of n-grams of the generated text with an original reference text and so only consider lexical similarity. Also, they require a lot of reference text to measure the quality of generation which is very costly to get. To overcome these challenges we propose a problem-solution alignment metric as an unsupervised evaluation measure.

5.1 Case Alignment

A key principle of CBR is that "*similar problems have similar solutions*". The extent to which this principle holds true can be assessed by measuring the alignment between the problem-side and solution-side space. It is surmised that a good system design will have better alignment [9]. In this evaluation, we employ a novel approach to measuring case alignment by using normalised discounted cumulative gain to assess the correlation between problem-side and solution-side nearest neighbours. If the alignment is good then for a given problem-solution pair, the k nearest cases on problem-side must be similar as the k nearest-cases on the solution side.

For a given case-base C containing all the cases $\{c_1, c_2, \cdots, c_n\}$. Cases in C consist of problem−solution pairs, such that $c_i = \{p_i, s_i\}$, where $p_i \in P$ (problem set) and $s_i \in S$ (solution set). A target problem t represented using the case knowledge, we will retrieve two lists pl & sl which are sorted in order to the most similar cases both from the problem (pl) and the solution (sl) set respectively. On the solution side, BERT [3] is used to encode the sentences and

then cosine similarity [15] between the test sample and other samples is used to generate the ordered list of similar cases. For the problem side the ordered list is created using the retrieval methods discussed in Sect. 4. Both the lists will have $n - 1$ items, where n is the size of the case-base.

From the list pl, we shall create a new list of weighted scores for the problem-side. We call it problem list weighted or plw. The weighting is done as follows:

$$plw(i) = \begin{cases} (k+1) - i, & \text{if } i \le k \\ 1, & \text{otherwise} \end{cases} \tag{3}$$

where k is the number of neighbours considered for retrieval and i is the index of each element from the pl. Similarly the cases in sl are weighted according to their pl counter-part and creating a solution list weighted or slw.

For example, if we have 10 cases in the case-base and for a given case c_i, with $k = 3$ the sl and pl are as follows:

$$pl = [8, 5, 6, 1, 4, 2, 3, 7, 0]$$
$$sl = [5, 6, 2, 4, 3, 7, 8, 1, 0]$$

These are the indices of the cases from both the sets. According to the Eq. (3), weighted lists plw and slw are given as follows:

$$plw = [4, 3, 2, 1, 1, 1, 1, 1, 1]$$
$$slw = [3, 2, 1, 1, 1, 1, 4, 1, 1]$$

For an ideal case, both of the list should have same ranking order as they are retrieved for the same case. To measure the alignment of a target case t we can use the *"normalised Discounted Cumulative Gain" (nDCG)* [16] using the following formula:

$$nDCG(t) = \frac{DCG_{(slw)}}{DCG_{(plw)}} \tag{4}$$

where, slw and plw are the weighted lists for the target case while DCG is the *"Discounted Cumulative Gain"*, defined for some list lw as:

$$DCG(lw) = \sum_{i=1}^{|lw|} \frac{lw(i)}{\log_2(i+1)} \tag{5}$$

where, lw is some weighted list (e.g., plw or slw) and $|lw|$ is the size of that list. The value of nDCG $\in (0, 1]$.

The alignment of whole case-base can be the average of nDCG score of all the cases in the case-base (CB).

$$AlignScore(CB) = \sum_{i=1}^{n} \frac{nDCG(i)}{n}, \forall i \in CB \tag{6}$$

where, n is the size of case-base. For component retrieval method, the total alignment score would be the average of *AlignScore* of all the components. In our experiments, we take the value of $k = 5$ because of the fact that we need to show 5 options of automatically generated obituaries to the user.

Table 1. Nomenclature of different methods.

	Basic	Component
$sim1$	**BS1**	**CS1**
$sim2$	**BS2**	**CS2**

5.2   Other Evaluations

In addition to the case alignment, we use BLEU score and cosine similarity for the evaluation of our system. BLEU score counts the average of overlapped n-grams from generated text with the reference texts. Cosine similarity on other hand measures the cosine angle between the projection of vectors in multi-dimensional space. For the vector representation of a sentence, we used BERT encoder to produce a contextual embedding for each sentence.

5.3   Average Attribute Errors

We define a reference metric as the number of missed attributes in the generated text as one measure of the competence of the evaluation metrics. In our scenario, where the pre-defined attributes play an important role in the retrieval and reuse of cases, it is important to measure the inclusion of these attributes in the generated text. In the absence of a human evaluation, we employ **Average Attribute Error (AAE)** as a crude surrogate for human feedback.

The average attribute error is defined as the average number of missed attributes from the top 5 generated texts from our system. Again, top 5 cases are chosen because of the fact that the system needs to provide 5 optional texts to the user for a given input. For a target problem t, if we have na number of attributes and the $G = \{g_1, \cdots, g_5\}$ as the set of top 5 generated texts from one of the methods defined in Table 1. The average attribute error e would be:

$$e(t) = \frac{\sum_{i=1}^{5} ||(na - |g_i|)||}{5} \tag{7}$$

where, $|g_i|$ is the number of attributes included in the ith generation. The average of every sample's attribute error in a case-base will be the average attribute error for the case-base.

6   Results and Discussion

Our case-base contains 100 seed cases manually annotated to identify problem and solution components. We use a leave-one-out experiment for both representations described in Sect. 4.3 (Basic and Component) with both similarity measures described in Sect. 4.2 ($sim1$ and $sim2$). Hence, We have four system combinations to evaluate as named in Table 1.
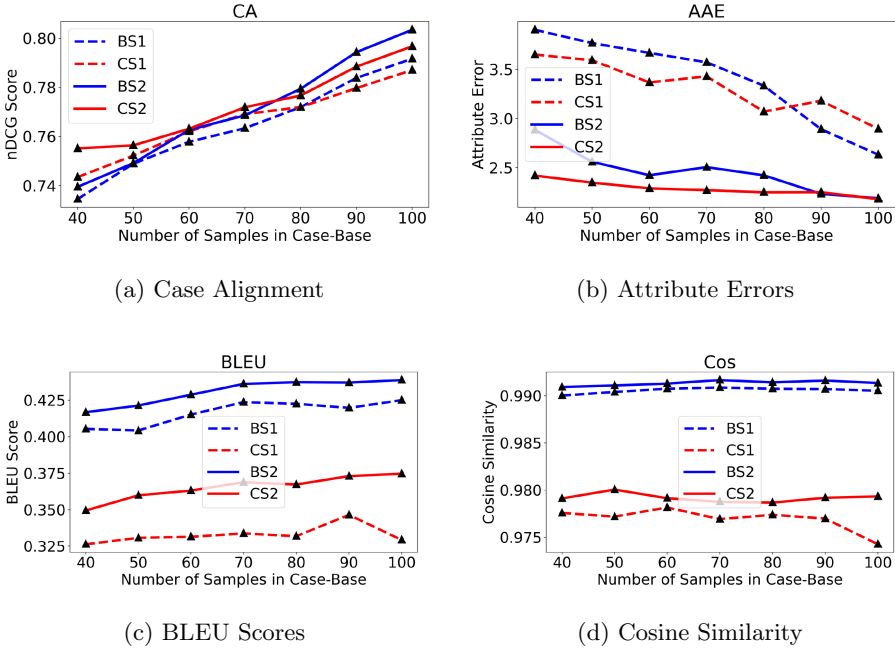
(a) Case Alignment

(b) Attribute Errors

(c) BLEU Scores

(d) Cosine Similarity

Fig. 3. Various results from leave-one-out-experiment

6.1 Different Evaluations

The results from applying the 4 evaluation metrics to the retrieval sets obtained when employing the 4 system combinations are shown in Fig. 3. We start our experiments with 40 cases initially, chosen to reflect the 40 attributes present in the problem representation. We repeat the experiments with increasing number of samples until we reach 100, i.e., the maximum number of seed cases available.

Case Alignment (CA) results are shown in Fig. 3a, where we plot the change in case alignment score with respect to the number of cases used for experiment. For a given value on the x-axis, the corresponding value on the y-axis represent the average case alignment score of all the cases from the leave-one-out experiment. We can see that with the number of samples increasing, the case alignment is also improving. Which means with more data used for experiment we are continuing to achieve improved results and do not appear to have reached a plateau.

We can also observe that before 70 cases the alignment is better for *component retrieval* compared to *basic retrieval* while after 70 samples, the *basic retrieval* for both *similarity measures* gets better alignment than the *component retrieval*. This indicates that after sufficient case data is available there may be no need to break down the obituary representation into several components because with more labelled cases, diversity in the case base is increased allowing sufficiently similar cases to be retrieved.

Table 2. Pearson coefficient score for correlation

	Case alignment	BLEU score	Cosine similarity
Pearson Score	−0.9238	−0.7019	−0.0296

Results from the **Average Attribute Error (AAE)** evaluation metric is shown in Fig. 3b. Here we can see that with the change in number of cases used for experiment, the average count of missing attributes is reducing for all the four system combinations. Also, before 80 cases, the performance for *component* is better, while after 80 *basic* for both *similarity measures* gives improved results. This further supports the idea that with more cases available, there is no need to split obituaries into components.

In Figs. 3c and 3d we show results for **BLEU score (BLEU)** and **cosine similarity (Cos)** between the generated text and reference text. Both the metrics show little variations in score with respect to the change in number of cases available. The BLEU score for BS1 and BS2 is always around 0.40 to 0.44 while for CS1 and CS2 is 0.33 to 0.37. Similarly for cosine similarity, the average is almost 0.99 for BS1 and BS2 during all the number of samples while the score for CS1 and CS2 is around 0.975 go 0.98. This may be because these metrics only consider lexical similarity while ignoring the measure of attributes inclusion for generation.

6.2   Correlation of Metrics

We calculate the pearson correlation coefficient between average attribute error and the other three automated evaluation metrics which is shown in Table 2. Here, we can observe that our proposed case alignment metric is highly correlated to the average attribute error. BLEU score is ranked second while the cosine similarity is third and is much less correlated. This demonstrates that our case alignment measure is an effective evaluation metric for the TCBR system.

6.3   Generated Texts

Some texts generated from the CBR system are shown in Fig. 4, one for each method from Table 1. The texts shown here are generated from the case-base with all 100 samples stored. It can be observed that the generations from $sim2$ measure are quite accurate and include most of the information correctly. On the other hand, it can be observed that the generations from $sim1$ measure have more tendency of making attribute's related errors such as: in BS1, the two major attributes, *age* and *charity_name* are missing; while in CS1, the retrieved case has some extra attributes which are not present in the target case, observe the sentence "Cherished to and a dear of the family". Here after *cherished to*, an extra attribute value is present in the retrieved case which is absent in the target problem and hence during the adaptation process, the value of that attribute is replaced with a blank.

Features (<*attribute*>: value)	Generations	
	BS1	**CS1**
<*age*>: 78 <*home_town*>: Cults <*name*>: John Doe <*gender*>: Male <*demise_date*>: 01 Feb 2020 <*demise_how*>: peacefully <*demise_place*>: Aberdeen Royal Infirmary <*children_in_law_name*>: Jacob <*children_name*>: Olivia <*friends_name*>: Lewis, Harris and Logan, <*grandchildren_name*>: Emily and James, <*grandparent_gender*>: papa <*parent_gender*>: father <*parent_in_law_gender*>: father in law <*spouse_gender*>: husband <*spouse_name*>: Jane Doe <*cemetery_place*>: Holburn Cemetery <*cemetery_time*>: 12:30 <*charity_name*>: British Heart Foundation <*guests_list*>: friends and family <*flowers*>: family flowers only <*funeral_attire*>: Blue and Grey kilts <*funeral_date*>: 05 Feb 2020 <*funeral_place*>: Trinity Church Aberdeen <*funeral_time*>: 10:00	John Doe (John) peacefully at Aberdeen Royal Infirmary on 01 February 2020, John much loved and devoted husband of Jane Doe, a dearly loved father of Olivia, father in law of Jacob, a doting papa to Emily and James and a friend to many. Our hearts are broken. Service at Trinity Church Aberdeen on 05 February 2020 at 10:00 burial thereafter to Holburn Cemetery arriving at 12:30 to which friends and family are respectfully invited. Missed Attributes: <*age*>, <*home_town*>, <*friends_name*>, <*funeral_attire*>, <*flowers*>, <*charity_name*>	John Doe peacefully at Aberdeen Royal Infirmary on 01 February 2020, , aged 78 years of Cults. . Beloved husband of Jane Doe, much loved father of Olivia. Loving father in law to Jacob. Devoted papa to Emily and James. Cherished to and a dear and of the family.. Funeral service to which friends and family are respectfully invited at Trinity Church Aberdeen on 05 February 2020 at 10:00, thereafter to Holburn Cemetery at 12:30. family flowers only please. A collection will be taken on retiral for British Heart Foundation. Missed Attributes: <*friends_name*>, <*funeral_attire*>
	BS2	**CS2**
	John Doe peacefully at Aberdeen Royal Infirmary on 01 February 2020, surrounded by his loving family, John Doe, aged 78 years. Beloved husband of Jane Doe, loving father to Olivia, cherished papa to Emily and James and father in law to Jacob. Sadly missed by all the family. Funeral service will be held at Trinity Church Aberdeen on 05 February 2020 at 10:00, to which friends and family are respectfully invited, followed by interment at Holburn Cemetery, arriving approximately 12:30. Family flowers only please as there will be a retiral collection for those wishing to donate to British Heart Foundation. Missed Attributes: <*home_town*>, <*friends_name*>, <*funeral_attire*>	John Doe peacefully at Aberdeen Royal Infirmary on 01 February 2020, John Doe aged 78 years of Cults. . devoted husband of Jane Doe. Treasured father of Olivia. Dearest father in law of Jacob and a cherished papa to Emily and James. Much loved and sadly missed by his family and many friends.. Funeral service will be held at Trinity Church Aberdeen on 05 February 2020 at 10:00, to which friends and family are respectfully invited, followed by interment at Holburn Cemetery, arriving approximately 12:30. family flowers only please as there will be a retiral collection for those wishing to donate to British Heart Foundation. Missed Attributes: <*friends_name*>, <*funeral_attire*>

Fig. 4. Text generations from the TCBR system. Errors are shown in red. (Color figure online)

It is also noted that missing the $< friends\_name >$ and $< funeral\_attire >$ attributes is common for all the four generations. We can also observe that the component retrieval method is also prone to different punctuation errors such as: ending the sentence with two full stops; or starting a sentence with small-caps letter. This may be due to the mix and match property of text generated from component retrieval methods.

7   Conclusion and Future Work

In this paper we presented a TCBR system developed for the automated generation of natural language obituaries from a large set of structured input attributes. The paper introduced two alternative case representation approaches, along with two different measures of similarity used for the retrieval of similar cases from the case-base.

The performance of our methods is evaluated using a novel unsupervised case alignment metric employing normalised discounted cumulative gain to compare problem-side and solution-side retrieval sets. Extensive experiments are conducted with an increasing number of seed cases available in a leave-one-out experiment. The proposed case alignment evaluation metric is compared with other commonly used supervised metrics as well as with average attribute error score, a simple surrogate for human feedback. The experiment results show that our unsupervised evaluation metric better correlates to the average attribute error compared to BLEU score and cosine similarity. Our evaluation metric is also domain independent and can be applied to different kinds of TCBR systems.

In future work for this project the intention is to measure and introduce more diversity into the set of generated obituaries and to automate the process of marking-up the data to ease the case-base creation process.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Commun. **7**(1), 39–59 (1994)
2. Adeyanju, I.: Generating weather forecast texts with case based reasoning. arXiv preprint arXiv:1509.01023 (2015)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
4. Dong, R., Schaal, M., O'Mahony, M.P., McCarthy, K., Smyth, B.: Harnessing the experience web to support user-generated product reviews. In: Agudo, B.D., Watson, I. (eds.) ICCBR 2012. LNCS (LNAI), vol. 7466, pp. 62–76. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32986-9_7
5. Dubey, N., Chakraborti, S., Khemani, D.: Textual summarization of time series using case-based reasoning: a case study. In: Workshop on Reasoning about Time in CBR-RATIC 2018. Workshop at the 26th International Conference on Case-Based Reasoning (ICCBR 2018), pp. 164–174 (2018)
6. Hüske-Kraus, D.: Text generation in clinical medicine - a review. Methods Inf. Med. **42**(1), 51–60 (2003)
7. Lamontagne, L.: Textual CBR authoring using case cohesion. In: Proceedings of 3rd Textual Case-Based Reasoning Workshop at the 8th European Conference on CBR, pp. 33–43 (2006)
8. Lin, C.Y.: ROUGE: a package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81. Association for Computational Linguistics, Barcelona, Spain, July 2004. https://www.aclweb.org/anthology/W04-1013
9. Massie, S., Wiratunga, N., Craw, S., Donati, A., Vicari, E.: From anomaly reports to cases. In: International Conference on Case-Based Reasoning, pp. 359–373 (2007)
10. Miura, N., Takagi, T.: WSL: sentence similarity using semantic distance between words. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pp. 128–131 (2015)
11. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318. Association for Computational Linguistics (2002)
12. Ramos-Soto, B.: Barro, Taboada: linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data. IEEE Trans. Fuzzy Syst. **23**(1), 44–57 (2015)
13. Recio-Garcıa, J.A., Dıaz-Agudo, B., González-Calero, P.A.: Textual CBR in JCOL-IBRI: from retrieval to reuse. In: Proceedings of the ICCBR 2007 Workshop on Textual Case-Based Reasoning: Beyond Retrieval, pp. 217–226 (2007)
14. Richter, M.M.: Knowledge containers. In: Readings in Case-Based Reasoning (2003)
15. Singhal, A., et al.: Modern information retrieval: a brief overview. IEEE Data Eng. Bull. **24**(4), 35–43 (2001)

16. Wang, Y., Wang, L., Li, Y., He, D., Liu, T.Y.: A theoretical analysis of NDCG type ranking measures. In: Conference on Learning Theory, pp. 25–54 (2013)
17. Zhou, X., Shi, Z., Zhao, H.: Reexamination of CBR hypothesis. In: Bichindaritz, I., Montani, S. (eds.) ICCBR 2010. LNCS (LNAI), vol. 6176, pp. 332–345. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14274-1_25