GenNext: A Consolidated Domain Adaptable NLG System

Frank Schilder, Blake Howald and Ravi Kondadadi*
Thomson Reuters, Research & Development
610 Opperman Drive, Eagan, MN 55123
firstname.lastname@thomsonreuters.com

Abstract

We introduce GenNext, an NLG system designed specifically to adapt quickly and easily to different domains. Given a domain corpus of historical texts, GenNext allows the user to generate a template bank organized by semantic concept via derived discourse representation structures in conjunction with general and domain-specific entity tags. Based on various features collected from the training corpus, the system statistically learns template representations and document structure and produces well–formed texts (as evaluated by crowdsourced and expert evaluations). In addition to domain adaptation, GenNext's hybrid approach significantly reduces complexity as compared to traditional NLG systems by relying on templates (consolidating micro-planning and surface realization) and minimizing the need for domain experts. In this description, we provide details of GenNext's theoretical perspective, architecture and evaluations of output.

1 Introduction

NLG systems are typically tailored to very specific domains and tasks such as text summaries from neonatal intensive care units (SUMTIME-NEONATE (Portet et al., 2007)) or offshore oil rig weather reports (SUMTIME-METEO (Reiter et al., 2005)) and require significant investments in development resources (e.g. people, time, etc.). For example, for SUMTIME-METEO, 12 person months were required for two of the system components alone (Belz, 2007). Given the subject matter of such systems, the investment is perfectly reasonable. However, if the domains to be generated are comparatively more general, such as financial reports or biographies, then the scaling of development costs becomes a concern in NLG.

NLG in the editorial process for companies and institutions where content can vary must be domain adaptable. Spending a year or more of development time to produce high quality market summaries, for example, is not a viable solution if it is necessary to start from scratch to produce other reports. GenNext, a hybrid system that statistically learns document and sentence template representations from existing historical data, is developed to be consolidated and domain adaptable. In particular, GenNext reduces complexity by avoiding the necessity of having a separate document planner, surface realizer, etc., and extensive expert involvement at the outset of system development.

Section 2 describes the theoretical background, architecture and implementation of GenNext. Section 3 discusses the results of a non–expert and expert crowdsourced sentence preference evaluation task. Section 4 concludes with several future experiments for system improvement.

2 Architecture of GenNext

In general, NLG systems follow a prototypical architecture where some input data from a given domain is sent to a "document planner" which decides content and structuring to create a document plan. That document plan serves as an input to a "micro planner" where the content is converted into a syntactic expression (with associated considerations of *aggregation* and *referring expression generation*) and a text specification is created. The text specification then goes through the final stage of "surface realization" where everything is put together into an output text (McKeown, 1985; Reiter and Dale, 2000; Bateman and Zock, 2003).

In contrast, the architecture of GenNext (summarized in Figure 1) is driven by a domain-specific

*Ravi Kondadadi is now affiliated with Nuance Communications, Inc.

Proceedings of the 14th European Workshop on Natural Language Generation, pages 178–182,
Sofia, Bulgaria, August 8-9 2013.

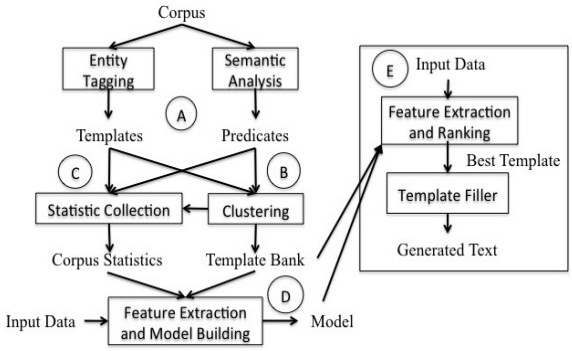

Figure 1: GenNext System Architecture.

corpus text. There is often a structured database underlying the domains of corpus text, the fields of which are used for domain specific entity tagging (in addition to domain general entity tagging [e.g. DATE, LOCATION, etc.]). An overview of the different stages, which are a combination of statistical (e.g., Langkilde and Knight (1998)) and template–based (e.g., van Deemter, et al. (2005)) approaches, follows in (A-E).[1]

A: Semantic Representation - We take a domain specific training corpus and reduce each sentence to a Discourse Representation Structure (DRS) - formal semantic representations of sentences (and texts) from Discourse Representation Theory (Kamp and Reyle, 1993; Basile and Bos, 2011). Each DRS is a combination of domain general named entities, predicates (content words) and relational elements (function words). In parallel, domain specific named entity tags are identified and are used to create templates that syntactically represent some conceptual meaning; for example, the short *biography* in (1):

(1) *Sentence*
a. Mr. Mitsutaka Kambe has been serving as Managing Director of the 77 Bank, Ltd. since June 27, 2008.
b. He holds a Bachelor's in finance from USC and a MBA from UCLA.

Conceptual Meaning
c. SERVING | MANAGING | DIRECTOR | PERSON | ...
d. HOLDS | BACHELOR | FINANCE | MBA | HOLD | ...

Once the semantic representations are created, they are organized and identified by semantic concept ("CuId") (described in (B)). Our assumption is that each cluster equates with a CuId represented by each individual sentence in the cluster and is contrastive with other CuIds (for similar approaches, see Barzilay and Lapata (2005), Angeli, et al. (2010) and Lu and Ng (2011)).

B: Creating Conceptual Units - To create the CuIds (a semi-automatic process), we cluster the sentences using *k*-means clustering with *k* set arbitrarily high to over-generate (Witten and Frank, 2005). This facilitates manual verification of the generated clusters to merge (rather than split) them if necessary. We assign a unique CuId to each cluster and associate each template in the corpus to a corresponding CuId. For example, in (2), using the sentences in (1a-b), the identified named entities are assigned to a clustered CuId (2a-b) and then each sentence in the training corpus is reduced to a template (2c-d).

(2) *Content Mapping*
a. $\{CuId : 000\}$ – *Information*: **person**: Mr. Mitsutaka Kambe; **title**: Managing Director; **company**: 77 Bank, Ltd.; **date**: June 27, 2008
b. $\{CuId : 001\}$ – *Information*: **person**: he; **degree**: Bachelor's, MBA; **subject**: finance; **institution**: USC; UCLA

Templates
c. $\{CuId : 000\}$: **[person]** has been serving as **[title]** of the **[company]** since **[date]**.
d. $\{CuId : 001\}$: **[person]** holds a **[degree]** in **[subject]** from **[institution]** and a **[degree]** from **[institution]**.

At this stage, we will have a set of CuIds with corresponding template collections which represent the entire "micro-planning" aspect of our system.

C: Collecting Statistics - For the "document planning" stage, we collect a number of statistics for each domain, for example:

- Frequency distribution of CuIds by position
- Frequency distribution of templates by position
- Frequency distribution of entity sequence
- Average number of entities by CuId and position

These statistics, in addition to entity tags and templates, are used in building different features used by the ranking model (D).

D: Building a Ranking Model - The core component of our system is a statistical model that ranks a set of templates for a given position (e.g. sentence 1, sentence 2, ..., sentence n) based on the input data (*see also* Konstas and Lapata (2012). The learning task is to find the rank for all the templates from all CuIds at each position. To generate the training data, we first exclude the templates that have named entities not specified in the input data (ensuring completeness). We then rank templates according to the edit distance (Levenshtein,

[1]For more detail see Howald, et al. (2013) - semantic clustering and micro-planning and Kondadadi, et al. (2013) - document planning.

1966) from the template corresponding to the current sentence in the training document. For each template, we build a ranking model with features, for example:

- Prior template and CuId
- Difference in number of words given position
- Most likely CuId given position and previous CuId
- Template 1-3grams given position and CuId

We use a linear kernel for a ranking SVM (Joachims, 2002) to learn the weights associated with each feature. Each domain has its own model that is used when generating texts (E).

E: Generation: At generation time, our system has a set of input data, a semantically organized template bank and a model from training on a given domain of texts. For each sentence, we first exclude those templates that contain a named entity not present in the input data. Then we calculate the feature values times the model weight for each of the remaining templates. The template with the highest score is selected, filled with matching entities from the input data and appended to the generated text. Example generations for each domain are included in (3).

(3) *Financial*
a. First quarter profit per share for Brown-Forman Corporation expected to be $0.91 per share by analysts.
b. Brown-Forman Corporation July first quarter profits will be below that previously estimated by Wall Street with a range between $0.89 and $0.93 per share and a projected mean per share of $0.91 per share.
c. The consensus recommendation is Hold.

Biography
d. Mr. Satomi Mitsuzaki has been serving as Managing Director of Mizuho Bank since June 27, 2008.
e. He was previously Director of Regional Compliance of Kyoto Branch.
f. He is a former Managing Executive Officer and Chief Executive Officer of new Industrial Finance Business Group in Mitsubishi Corporation.

Weather
g. Complex low from southern Norway will drift slowly NNE to the Lofoten Islands by early tomorrow.
h. A ridge will persist to the west of British Isles for Saturday with a series of weak fronts moving east across the North Sea.
i. A front will move ENE across the northern North Sea Saturday.

3 Evaluation and Discussion

We have tested GenNext on three domains: Corporate Officer and Director Biographies (1150 texts ranging from 3-10 period ended sentences), Financial Texts (Mutual Fund Performances [162 texts, 2-4 sentences] and Broker Recommendations [905 texts, 8-20 sentences]), and Offshore Oil Rig Weather Reports (1054 texts, 2-6 sentences) from SUMTIME-METEO (Reiter et al., 2005). The total number of templates for the *financial* domain is 1379 distributed across 38 different semantic concepts; 2836 templates across 19 concepts for *biography*; and 2749 templates across 9 concepts for *weather* texts.

We have conducted several evaluation experiments comparing two versions of GenNext, one applying the ranking model (*rank*) and one with random selection of templates (*non-rank*) (both systems use the same template bank, CuId assignment and filtering) and the original texts from which the data was extracted (*original*).

We used a combination of automatic (e.g. BLEU–4 (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2011)) and human metrics (using crowdsourcing) to evaluate the output (*see generally*, Belz and Reiter (2006). However, in the interest of space, we will restrict the discussion to a human judgment task on output *preferences*. We found this evaluation task to be most informative for system improvement. The task asks an evaluator to provide a binary preference determination (100 sentence pairs/domain): "Do you prefer Sentence A (from *original*) or the corresponding Sentence B (from *rank* or *non-rank*)". This task was performed for each domain.[2] We also engaged 3 experts from the financial and 4 from the biography domains to perform the same preference task (average agreement was 76.22) as well as provide targeted feedback.

For the preference results, summarized in Figure 2, we would like to see no statistically significant difference between GenNext-*rank* and *original*, but statistically significant differences between GenNext-*rank* and GenNext-*non-rank*, and *original* and GenNext-*non-rank*. If this is the case, then GenNext-*rank* is producing texts similar to the *original* texts, and is providing an observable improvement over not including the model at all (GenNext-*non-rank*). This is exactly what we see for all domains.[3] However, in general, there

[2]Over 100 native English speakers contributed, each one restricted to providing no more than 50 responses and only after they successfully answered 4 initial gold data questions correctly and continued to answer periodic gold data questions. The pair orderings were randomized to prevent click bias. 8 judgments per sentence pair was collected (2400 judgments) and average agreement was 75.87.

[3]*Original* vs. GenNext-*rank* : *financial* - χ^2=.29, $p\leq$.59; *biography* - χ^2=3.01, $p\leq$.047; *weather* - χ^2=.95, $p\leq$.32. *Original* vs. GenNext-*non-rank* : *financial* - χ^2=16.71, $p\leq$.0001; *biography* - χ^2=45.43, $p\leq$.0001; *weather* -

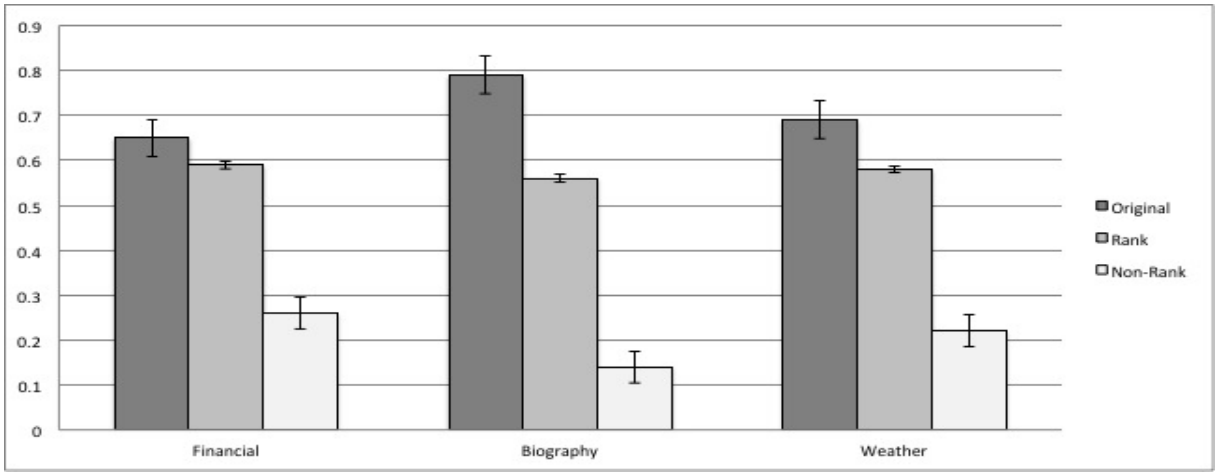

Figure 2: Cross-Domain Non-Expert Preference Evaluations.

is a greater difference between the *original* and GenNext-*rank biographies* compared to the *financial* and *weather* texts. We take it as a goal to approach, as close as possible, the preferences for the *original* texts.

The original *financial* documents were machine generated from a different existing system. As such, it is not surprising to see similarity in performance compared to GenNext-*rank* and potentially explains why preferences for the originals is somewhat low (assuming a higher preference rating for well-formed human texts). Further, the original *weather* documents are highly technical and not easily understood by the lay person, so, again, it is not surprising to see similar performance. *Biographies* were human generated and easy to understand for the average reader. Here, both GenNext-*rank* and GenNext-*non-rank* have some ground to make up. Insights from domain experts are potentially helpful in this regard.

Expert evaluations provided similar results and agreements compared to the non–expert crowd. Most beneficial about the expert evaluations was the discussion of integrating certain editorial standards into the system. For example, shorter texts were preferred to longer texts in the *financial* domain, but not the *biographies*. Consequently, we could adjust weights to favor shorter templates. Also, in *biographies*, sentences with subordinated elaborations were not preferred because these contained subjective comments (e.g. *a leader in industry*, *a well respected individual*, etc.). Here, we could manually curate or could automatically detect templates with subordinated clauses and remove them. These types of comments are useful to adjust the system accordingly to end user expectations.

4 Conclusion and Future Work

We have presented our system GenNext which is domain adaptable, given adequate historical data, and has a significantly reduced complexity compared to other NLG systems (*see generally*, Robin and McKeown (1996)). To the latter point, development time for semantically processing the corpus, applying domain general and specific tags, and building a model is accomplished in days and weeks as opposed to months and years.

Future experimentation will focus on being able to automatically extract templates for different domains to create preset banks of templates in the absence of adequate historical data. We are also looking into different ways to increase the variability of output texts from selecting templates within a range of top scores (rather than just the highest score) to providing additional generated information from input data analytics.

Acknowledgments

This research is made possible by Thomson Reuters Global Resources (TRGR) with particular thanks to Peter Pircher, Jaclyn Sprtel and Ben Hachey for significant support. Thank you also to Khalid Al-Kofahi for encouragement, Leszek Michalak and Andrew Lipstein for expert evaluations and three anonymous reviewers for constructive feedback.

χ^2=24.27, $p\leq$.0001. GenNext-*rank* vs. GenNext-*non-rank* : *financial* - χ^2=12.81, $p\leq$.0003; *biography* - χ^2=25.19, $p\leq$.0001; *weather* - χ^2=16.19, $p\leq$.0001.

References

Gabor Angeli, Percy Liang, and Dan Klein. 2012. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods for Natural Language Processing (EMNLP 2010)*, pages 502–512.

Regina Barzilay and Mirella Lapata. 2005. Collective content selection for concept-to-text generation. In *Proceedings of the 2005 Conference on Empirical Methods for Natural Language Processing (EMNLP 2005)*, pages 331–338.

Valerio Basile and Johan Bos. 2011. Towards generating text from discourse representation structures. In *Proceedings of the 13th European Workshop on Natural Language Generation (ENLG)*, pages 145–150.

John Bateman and Michael Zock. 2003. Natural language generation. In R. Mitkov, editor, *Oxford Handbook of Computational Linguistics*, Research in Computational Semantics, pages 284–304. Oxford University Press, Oxford.

Anja Belz and Ehud Reiter. 2006. Comparing automatic and human evaluation of NLG systems. In *Proceedings of the European Association for Computational Linguistics (EACL'06)*, pages 313–320.

Anja Belz. 2007. Probabilistic generation of weather forecast texts. In *Proceedings of Human Language Technologies 2007: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'07)*, pages 164–171.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, pages 85–91.

Blake Howald, Ravi Kondadadi, and Frank Schilder. 2013. Domain adaptable semantic clustering in statistical NLG. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)*, pages 143–154. Association for Computational Linguistics, March.

Thorsten Joachims. 2002. *Learning to Classify Text Using Support Vector Machines*. Kluwer.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic; An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer, Dordrecht.

Ravi Kondadadi, Blake Howald, and Frank Schilder. 2013. A statistical NLG framework for aggregated planning and realization. In *Proceedings of the Annual Conference for the Association of Computational Linguistics (ACL 2013)*. Association for Computational Linguistics.

Ioannis Konstas and Mirella Lapata. 2012. Concept-to-text generation via discriminative reranking. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 369–378.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL'98)*, pages 704–710.

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10:707–710.

Wei Lu and Hwee Tou Ng. 2011. A probabilistic forest-to-string model for language generation from typed lambda calculus expressions. In *Proceedings of the 2011 Conference on Empirical Methods for Natural Language Processing (EMNLP 2011)*, pages 1611–1622.

Kathleen R. McKeown. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press.

Kishore Papineni, Slim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, pages 311–318.

Franois Portet, Ehud Reiter, Jim Hunter, and Somayajulu Sripada. 2007. Automatic generation of textual summaries from neonatal intensive care data. In *In Proccedings of the 11th Conference on Artificial Intelligence in Medicine (AIME 07). LNCS*, pages 227–236.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press.

Ehud Reiter, Somayajulu Sripada, Jim Hunter, and Jin Yu. 2005. Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167:137–169.

Jacques Robin and Kathy McKeown. 1996. Empirically designing and evaluating a new revision-based model for summary generation. *Artificial Intelligence*, 85(1-2).

Kees van Deemter, Mariët Theune, and Emiel Krahmer. 2005. Real *vs.* template-based natural language generation: a false opposition? *Computational Linguistics*, 31(1):15–24.

Ian Witten and Eibe Frank. 2005. *Data Mining: Practical Machine Learning Techniques with Java Implementation (2nd Ed.)*. Morgan Kaufmann, San Francisco, CA.