Aligning Texts and Knowledge Bases with Semantic Sentence Simplification

Yassine Mrabet[1,3], Pavlos Vougiouklis[2], Halil Kilicoglu[1],
Claire Gardent[3], Dina Demner-Fushman[1], Jonathon Hare[2], and Elena Simperl[2]

[1]Lister Hill National Center for Biomedical Communications
National Library of Medicine, USA
{mrabety,kilicogluh,ddemner}@mail.nih.gov

[2]Web and Internet Science Research Group
University of Southampton, UK
{pv1e13,jsh2,es}@ecs.soton.ac.uk

[3]CNRS/LORIA, France
claire.gardent@loria.fr

Abstract

Finding the natural language equivalent of structured data is both a challenging and promising task. In particular, an efficient alignment of knowledge bases with texts would benefit many applications, including natural language generation, information retrieval and text simplification. In this paper, we present an approach to build a dataset of triples aligned with equivalent sentences written in natural language. Our approach consists of three main steps. First, target sentences are annotated automatically with knowledge base (KB) concepts and instances. The triples linking these elements in the KB are extracted as candidate facts to be aligned with the annotated sentence. Second, we use textual mentions referring to the subject and object of these facts to semantically simplify the target sentence via crowdsourcing. Third, the sentences provided by different contributors are post-processed to keep only the most relevant simplifications for the alignment with KB facts. We present different filtering methods, and share the constructed datasets in the public domain. These datasets contain 1,050 sentences aligned with 1,885 triples. They can be used to train natural language generators as well as semantic or contextual text simplifiers.

1 Introduction

A large part of the information on the Web is contained in databases and is not suited to be directly accessed by human users. A proper exploitation of these data requires relevant visualization techniques which may range from simple tabular presentation with meaningful queries, to graph generation and textual description. This last type of visualization is particularly interesting as it produces an additional raw resource that can be read by both computational agents (e.g. search engines) and human users. From this perspective, the ability to generate high quality text from knowledge and data bases could be a game changer.

In the Natural language Processing community, this task is known as Natural Language Generation (NLG). Efficient NLG solutions would allow displaying the content of knowledge and data bases to lay users; generating explanations, descriptions and summaries from ontologies and linked open data[1]; or guiding the user in formulating knowledge-base queries.

However, one strong and persistent limitation to the development of adequate NLG solutions for the semantic web is the lack of appropriate datasets on which to train NLG models. The difficulty is that the semantic data available in knowledge and data bases need to be aligned with the corresponding text. Unfortunately, this alignment task is far from straightforward. In fact, both human beings and machines perform poorly on it.

[1]http://www.linkeddata.org

Proceedings of the 2nd International Workshop on Natural Language Generation and the Semantic Web (WebNLG), pages 29–36,
Edinburgh, Scotland, September 6th, 2016.

Nonetheless, there has been much work on data-to-text generation and different strategies have been used to create the data-to-text corpora that are required for learning and testing. Two main such strategies can be identified. One strategy consists in creating a small, domain-specific corpus where data and text are manually aligned by a small group of experts (often the researchers who work on developing the NLG system). Typically, such corpora are domain specific and of relatively small size while their linguistic variability is often restricted.

A second strategy consists in automatically building a large data-to-text corpus in which the alignment between data and text is much looser. For instance, Lebret et al. (2016) extracted a corpus consisting of 728,321 biography articles from English Wikipedia and created a data-to-text corpus by simply associating the infobox of each article with its introduction section. The resulting dataset has a vocabulary of 403k words but there is no guarantee that the text actually matches the content of the infobox.

In this paper, we explore a middle-ground approach and introduce a new methodology for semi-automatically building large, high quality data-to-text corpora. More precisely, our approach relies on a semantic sentence simplification method which allows transforming existing corpora into sentences aligned with KB facts. Contrary to manual methods, our approach does not rely on having a small group of experts to identify alignments between text and data. Instead, this task is performed (i) by multiple, independent contributors through a crowdsourcing platform, and (ii) by an automatic scoring of the quality of the contributions, which enables faster and more reliable data creation process. Our approach also departs from the fully automatic approaches (e.g., (Lebret et al., 2016)) in that it ensures a systematic alignment between text and data.

In the following section we present work related to corpus generation for NLG. In section 3 we describe our approach. Section 4 presents the experiments, evaluations, and the statistics on the initial corpora and the generated (aligned) datasets.

2 Related Work

Many studies tackled the construction of datasets for natural language generation. Several available datasets were created by researchers and developers working on NLG systems. Chen and Mooney (2008) created a dataset of text and data describing the Robocup game. To collect the data, they used the Robocup simulator (www.robocup.org) and derived symbolic representations of game events from the simulator traces using a rule-based system. The extracted events are represented as atomic formulas in predicate logic with timestamps. For the natural language portion of the data, they had humans comment games while watching them on the simulator. They manually aligned logical formulas to their corresponding sentences. The resulting data-to-text corpus contains 1,919 scenarios where each scenario consists of a single sentence representing a fragment of a commentary on the game, paired with a set of logical formulas.

The SumTime-Meteo corpus was created by the SumTime project (Sripada et al., 2002). The corpus was collected from the commercial output of five different human forecasters, and each instance in the corpus consists of three numerical data files produced by three different weather simulators, and the weather forecast file written by the forecaster. To train a sentence generator, (Belz, 2008) created a version of the SumTime-Meteo corpus which is restricted to wind data. The resulting corpus consists of 2,123 instances for a total of 22,985 words and was used by other researchers working on NLG and semantic parsing (Angeli et al., 2012).

Other data-to-text corpora were proposed for training and testing generation systems, including WeatherGov (Liang et al., 2009), the ATIS dataset, the Restaurant Corpus (Wen et al., 2015) and the BAGEL dataset (Mairesse et al., 2010). WeatherGov consists of 29,528 weather scenarios for 3,753 major US cities. In the air travel domain, the ATIS dataset (Dahl et al., 1994) consists of 5,426 scenarios. These are transcriptions of spontaneous utterances of users interacting with a hypothetical online flight-booking system. The RESTAURANTS corpus contains utterances that a spoken dialogue system might produce in an interaction with a human user together with the corresponding dialog act. Similarly, the BAGEL dataset is concerned with restaurant information in a dialog setting.

In all these approaches, datasets are created using heuristics often involving extensive manual labour and/or programming. The data is

mostly created artificially from sensor or web data rather than extracted from some existing knowledge base. As the data are often domain specific, the vocabulary size and the linguistic variability of the target text are often restricted.

Other approaches tackled the benchmarking of NLG systems and provided the constructed dataset as a publicly available resource. For instance, a Surface Realisation shared task was organised in 2011 to compare and evaluate sentence generators (Belz et al., 2011). The dataset prepared by the organisers was derived from the PennTreebank and associates sentences with both a shallow representation (dependency trees) and a deep representation where edges are labelled with semantic roles (e.g., agent, patient) and the structure is a graph rather than a tree. While the data-to-text corpus that was made available from this shared task was very large, the representation associated with each sentence is a linguistic representation and is not related to a data schema.

The KBGen shared task (Banik et al., 2013) followed a different approach and focused on generating sentences from knowledge bases. For this task, knowledge base fragments were extracted semi-automatically from an existing biology knowledge base (namely, BioKB101 (Chaudhri et al., 2013)) and expert biologists were asked to associate each KB fragments with a sentence verbalising their meaning. The resulting dataset was small (207 data-text instances for training, 70 for testing) and the creation process relied heavily on domain experts, thereby limiting its portability.

In sum, there exists so far no standard methodology for rapidly creating data-to-text corpora that are both sufficiently large to support the training and testing of NLG systems and sufficiently precise to support the development of natural language generation approaches that can map KB data to sentences. The procedures designed by individual researchers to test their own proposals yield data in non-standard formats (e.g., tabular information, dialog acts, infoboxes) and are often limited in size. Data used in shared tasks either fail to associate sentences with knowledge base data (SR shared task) or require extensive manual work and expert validation.

3 Methods

Our approach tackles the conversion of existing textual corpora into a dataset of sentences aligned with <subject, predicate, object> triples collected from existing KBs. It is independent from the selected corpus, domain, or KB.

In the first step, we annotate automatically the target textual corpus by linking textual mentions to knowledge base concepts and instances (KB entities for short). In the second step, we collect triples from the knowledge bases that link the entities mentioned in a given sentence. In the third step, we keep only the mentions that refer to the subject and object of the same triple and perform semantic simplification with a crowdsourcing task. Finally we apply several post-processing algorithms, including clustering and scoring to keep the most relevant semantic simplifications of each sentence as a natural language expression of the set of collected triples.

The alignment that we aim to achieve is not binary, as an output of our approach, one sentence could be aligned with N triples ($N \geq 1$). This property is particularly interesting for NLG as it allows training generation systems on expressing sets of triples in the same sentence; enabling the production of more fluent texts.

3.1 Corpus Annotation and Initial Sentence Selection

In the following we present our methods to obtain automatic initial annotations of the target corpora and to select the sentences that will be used in the final aligned dataset.

3.1.1 Corpus Annotation

In order to have varied empirical observations, we use two different methods for initial corpus annotation. In the **first annotation method** we do not check if the candidate triples are actually expressed in the sentence, only their subjects and objects. This method is particularly suitable to discover new linguistic expressions of triple predicates, and can provide actual expressions of the triple by accumulating observations from different sentences.

To implement this method we use KODA (Mrabet et al., 2015) to link textual mentions to KB entities. KODA is an unsupervised entity linking tool that relies only on the KB contents to detect and disambiguate textual mentions. More precisely, it detects candidate textual mentions with a TF-IDF search on the labels of KB entities, and disambiguates them by maximizing the coherence between the candidate KB entities retrieved for each

mention using KB relations.

In the second step we query the KB (e.g., SPARQL endpoint of DBpedia) to obtain the predicates that link the KB entities mentioned in the sentence and keep them as candidate facts. For instance, the 8 highlighted terms in figure 1 were linked to DBpedia entities, but only 4 terms mention KB entities that are linked in DBpedia triples.

This first method is scalable w.r.t. the domain of interest as it can be ported to other KBs with the same implementation.

In the **second annotation method**, we perform the automatic annotation by checking that the triples are actually expressed in the sentence. We use SemRep (Rindflesch and Fiszman, 2003), a biomedical relation extraction system. SemRep extracts binary relations from unstructured texts. The subject and object of these relations are concepts from the UMLS Metathesaurus (Lindberg et al., 1993) and the predicate is a relation type from an expanded version of the UMLS Semantic Network (e.g., *treats*, *diagnoses*, *stimulates*, *inhibits*). SemRep uses MetaMap (Aronson and Lang, 2010) to link noun phrases to UMLS Metathesaurus concepts. For example, the 4 highlighted terms in figure 2 were linked to UMLS concepts and all terms mention either the subject or the object of a relation extracted with SemRep.

In both methods, we keep only the annotations that refer to subjects and objects of candidate facts.

3.1.2 Initial Sentence Selection.

Due to the unsupervised aspect of automatic annotation and the incompleteness of the KBs, some sentences are expected to be annotated more heavily than others, and some sentences are expected to have more triples associated with them than others. In practice, different targets of annotation (e.g. specific semantic categories) could also lead to similar discrepancies.

In order to train automatic sentence simplifiers with our datasets, we have to consider different levels of coverage that can correspond to different annotation tools and dissimilar annotation goals. Accordingly, once the initial corpus is annotated, we select three sets of sentences: (1) a first set of sentences that are *heavily annotated* w.r.t. the number of triples (e.g. between 5 and 10 tokens per triple), (2) a second set with average annotation coverage (e.g. between 10 and 20 tokens per triple), and (3) a third set of weakly annotated sentence (e.g. above 20 tokens per triple).

3.2 Semantic Sentence Simplification (S3)

In order to obtain the final dataset of KB facts aligned with natural language sentences from the initial automatically annotated corpus, we define the task of Semantic Sentence Simplification (S3) and introduce the crowdsourcing process used to perform it.

Definition. Given a sentence S, a set of textual mentions $M(S)$ linked to a set of KB instances and concepts $E(S)$ and a set of triples $T(S) = \{t_i(e_{i_1}, p_i, e_{i_2}), s.t. e_1 \in E(S), e_2 \in E(S)$, the semantic simplification task consists of *shortening the sentence* S as much as possible according to the following rules:

- Keep the textual mentions referring to the subject and object of candidate facts.
- Keep the relations expressed between these textual mentions in the sentence.
- Keep the order of the words from the original sentence as much as possible.
- Ensure that the simplified sentence is grammatical and meaningful.
- Avoid using external words to the extent possible.

Crowdsourcing. We asked contributors to provide simplifications for each sentence through a crowdsourcing platform. We highlighted the textual mentions referring to subjects and objects of candidate facts in these sentences. The contributors are then asked to follow the S3 requirements to shorten the sentences. The quality requirement that was set during the experiment is that each contributor should dedicate at least 15 seconds for each set of 3 sentences.

After several preliminary experiments, we opted for a crowdsourcing process without quiz questions to attract more participants; and we monitored closely the process to filter out irrelevant contributors such as spammers (e.g. people typing in random letters), foreign-language speakers who misunderstood the task and tried to provide translations of the original sentence, and contributors who simply copied the original sentence. By flagging such contributors we also optimized significantly the monitoring for the second corpus.

***Sacco** flew as a **payload specialist** on **STS-73**, which launched on **October 20, 1995**, and landed at the **Kennedy Space Center** on **November 5, 1995**.*

Mention	DBpedia Entity
Sacco	dbr:Albert_Sacco
payload specialist	dbr:Payload_Specialist
STS-73	dbr:STS-73
October 20	dbr:October_20
1995	dbr:1995
Kennedy Space Center	dbr:Kennedy_Space_Center
November 5	dbr:November_5

Triples

dbr:Albert_Sacco	dbo:mission	dbr:STS-73
dbr:STS-73	dbp:landingSite	dbr:Kennedy_Space_Center
dbr:STS-73	dbp:launchSite	dbr:Kennedy_Space_Center

Figure 1: Example sentence annotated with DBpedia entities and its candidate triples.

*The **antiviral agent amantadine** has been used to manage **Parkinson's disease** or **levodopa-induced dyskinesias** for nearly 5 decades.*

Mention	UMLS Entity
amantadine	C0002403
antiviral agent	C0003451
Parkinson's disease	C0030567
levodopa-induced dyskinesias	C1970038

Triples

Amantadine	*isa*	Antiviral Agents
Amantadine	*treats*	Parkinson Disease
Amantadine	*treats*	Levodopa-induced dyskinesias

Figure 2: Example sentence annotated with UMLS concepts and triples.

3.3 Selecting the best simplification

In order to select the most relevant simplification for a given sentence from the set of N simplifications proposed by contributors, we test two baseline methods and two advanced scoring methods.

3.3.1 Baselines.

The *first baseline method* is simply the selection of the simplification that has more votes. We will refer to it as $Vote$ in the remainder of the paper. The *second baseline method*, called $Clustering$, is based on the K-Means clustering algorithm. It uses the Euclidean distance measured between word vectors to cluster the set of N simplifications of a given sentence into K clusters. The cluster with the highest cumulative number of votes is selected as the most significant cluster, and the shortest sentence in that cluster is selected as the candidate simplification.

3.3.2 Scoring Methods

Our first selection method scores a simplification according to the original sentence and to the simplification goals expressed in section 3.3. We define four elementary measures to compute a semantic score: *lexical integrity*, *semantic preservation*, *conformity* and *relative shortening*. Given an initial sentence s_o and a simplification s_i proposed for s_o, these measures are defined as follows.

Conformity (cnf). The conformity score represents how much the simplification s_i conforms to the rules of the S3 task. It combines lexical integrity and semantic preservation:

$$cnf(s_i, s_o) = \zeta(s_i, s_o) \times \iota(s_i, s_o) \qquad (1)$$

Lexical integrity (ι). $\iota(s_i, s_o)$ is the proportion of words in s_i that are in s_o. ι values are in the [0,1] range. The value is lower than 1 if new external words are used.

Semantic Preservation (ζ). Semantic preservation indicates how much of the textual mentions that are linked to KB entities and KB triples are present in the simplification. More precisely, $\zeta(s_i, s_o)$ is the ratio of annotations from s_o that are present in s_i. ζ values are in the [0,1] range.

Relative Shortening (η). Simplifications that are too short might miss important relations or entities, whereas simplifications that are too long

might be too close (or equal) to the original sentence. We represent both aspects through a Gaussian and make use of the "wisdom of the crowd" by setting the maximum value at the average length of the simplifications proposed by the contributors. In order to have a moderate decrease around the average, we set both the maximum value and the standard deviation to 1. Length is measured in terms of tokens.

$$\eta(s_i, s_o) = exp(-\frac{(length(s_i) - length_{avg})^2}{2}) \quad (2)$$

Semantic score (ψ). We compute the semantic score for a simplification s_i of s_o by combining the above elements. This combination, expressed in equation 3, is based on the following intuitions: (1) between two simplifications of the same sentence, the difference in conformity should have more impact than the difference in shortening, (2) for the same conformity value, simplifications that are farther from the original sentence are preferred, and (3) simplifications that have a more common shortening extent should be better ranked.

$$\psi(s_i, s_o) = \eta(s_i, s_o) \times exp(cnf(s_i, s_o) \times euclidean(s_i, s_o)) \quad (3)$$

The Euclidean function is the Euclidean distance between the original sentence and the simplification in terms of tokens. Our *second scoring method* relied first on the clustering of the contributors' sentences. As the baseline it identifies the cluster with more votes as most significant. However, the representative sentence is selected according to the semantic score ψ, instead of simply taking the shortest sentence of the cluster. We denote this in-cluster scoring method ξ.

4 Experiments and Results

In the first experiments, we build two datasets of natural language sentences aligned with KB facts.

Corpora and knowledge bases. Our *first dataset* is built by aligning all astronaut pages on Wikipedia[2] (Wiki) with triples from DBpedia[3]. The main motivation behind the choice of this corpus is to have both general and specific relations. We used KODA as described in section 3.1.1 to obtain initial annotations.

Our *second dataset* is built by aligning the medical encyclopedia part of Medline Plus[4] (MLP) with triples extracted with SemRep. The motivation behind the selection of this corpus is twofold: a) to experiment with a domain-specific corpus, and b) to test the simplification when the triples are extracted from the text itself. Table 1 presents raw statistics on each corpus.

	Wiki	**MLP**
Documents	668	4,360
Sentences	22,820	16,575
Tokens	478,214	421,272
Token per Sentence	20.95	25.41
Triples	15,641	30,342
Triples per Sentence	0.68	1.83
Mentions	64,621	145,308
Arguments	13,751	47,742

Table 1: Basic Statistics on Initial Corpora

Crowdsourcing. We used CrowdFlower[5] as a crowdsourcing platform. We submitted 600 annotated sentences for the Wiki corpus and 450 sentences for the MLP corpus.

Selection of relevant simplifications. We implemented several methods to select the best simplification among the 15 contributions for each sentence (cf. section 3.3). To evaluate these methods we randomly selected 90 initial sentences from each dataset, then extracted the best simplification according to each of the 4 scoring metrics. The authors then rated each simplification from 1 to 5, with 1 indicating a very bad simplification, and 5 indicating an excellent simplification. One of the authors prepared the evaluation tables, anonymized the method names and did not participate in the evaluation. The remaining 6 authors shared the 180 sentences and performed the ratings. Table 2 presents the final average rating for each selection method.

	Baselines		Scoring	
	Vote	Clustering	ξ	ψ
Wiki	**3.62**	3.06	3.51	3.22
MLP	3.51	2.87	**3.61**	3.30
Overall	**3.56**	2.97	**3.56**	3.26

Table 2: Evaluation of S3 Selection Methods (average rating)

Final statistics on aligned datasets. After evalu-

[2] https://en.wikipedia.org/wiki/List_of_astronauts_by_name
[3] http://dbpedia.org
[4] https://www.nlm.nih.gov/medlineplus/
[5] http://www.crowdflower.com

	Wiki	MLP
Sentences	600	450
Triples	1,119	766
Predicates	146	30
Tokens	13,641 (after S3: **9,011**)	11,167 (after S3: **6,854**)

Table 3: Statistics on the Aligned Dataset

ating the selection methods we selected the most relevant simplification for each sentence in the dataset according to ξ (i.e., in-cluster scoring), and generated the final datasets that link the best simplification to the facts associated with its original sentence. Table 3 presents the final statistics on the aligned datasets. Both datasets are made available online[6].

Table 4 presents the 10 first predicate names and their distribution for each dataset.

Wiki		MLP	
Predicate	%	Predicate	%
rdf:type	15.6	location of	24.93
dbo:type	10.18	is a	20.75
dbo:mission	9.11	process of	14.09
dbo:crewMembers	6.34	treats	7.04
dbo:birthPlace	5.45	causes	6.78
dbo:occupation	4.64	part of	5.87
dbo:nationality	3.30	administred to	3.13
dbo:rank	3.03	coexists with	2.61
dbp:crew2Up	2.94	affects	2.08
dbo:country	1.96	uses	1.43

Table 4: Top 10 predicates

5 Discussion

Automatic Annotation. From our observations on both datasets, we came to the conclusion that uncertainty is required to some extent in the selection of candidate triples. This is due to the fact that relations extracted from the text itself will follow the patterns that were used to find them (e.g., regular expressions, or classifier models) and that will not allow finding enough variation to enrich NLG systems. From this perspective, the best option would be to rank candidate triples according to their probability of occurrence in the sentence and filter out the triples with very low probability. This ranking and filtering are planned for the final version of our open-domain corpus.

Initial sentence selection. The second goal of our datasets is to be able to train automatic semantic simplifiers that would reduce the need for manual simplification in the long term. Therefore, our first method took into account different levels of annotation coverage in order to cope with different performance/coverage of annotation tools and dissimilar goals in terms of the semantic categories of the mentions. However, for NLG, it is also important to have a balanced number of samples for each unique predicate. The first extension of our datasets will provide a better balance of examples for each predicate while keeping the balance in terms of annotation coverage to the extent possible.

Crowdsourcing. Our crowdsourcing experiment showed that it is possible to obtain relevant semantic simplifications with no specific expertise. This is supported by the fact that the $Vote$ baseline in the selection of the final simplification obtained the same best performance as our scoring method that relies on the semantics of the S3 process. Overall, the experiment cost was only $180 for 15,750 simplifications collected for 1,050 sentences. Our results also show that collecting only 10 simplifications for each sentence (instead of 15 in our experiments) would be more than adequate, which reduces the costs even further. The two jobs created for each dataset were generally well-rated by the contributors (cf. Table 5). The MLP corpus was clearly more accessible than the Wiki corpus with an ease of job estimated at 4.4 vs 3.8 (on a 5 scale). Interestingly, the identical instructions were also rated differently according to the dataset (4.2 vs. 3.8). The Wiki corpus was harder to process, due to the high heterogeneity of the relations and entity categories. There are also fewer arguments per sentence in the Wiki corpus: 0.68 triple per sentence vs. 1.83, for a close average length of 20.95 tokens per sentence vs. 25.41 (cf. Table 1).

	Wiki	MLP
Number of participants	48	41
Clarity of Instructions	3.8	4.2
Ease of Job	3.8	4.4
Overall Rating of Job	3.9	4.4

Table 5: Number of participants and contributors' ratings (on a 1 to 5 scale)

[6] https://github.com/pvougiou/KB-Text-Alignment

6 Conclusions

We presented a novel approach to build a corpus of natural language sentences aligned with knowledge base facts, and shared the first constructed datasets in the public domain. We introduced the task of semantic sentence simplification that retains only the natural language elements that correspond minimally to KB facts. While our simplification method relied on crowdsourcing, our midterm goal is to collect enough data to train automatic simplifiers that would perform the same task efficiently. Besides the simplification aspect and the portability of the method, the shared datasets are also a valuable resource for natural language generation systems. Future work includes the expansion of these datasets and the improvement of sentence selection using grammatical-quality factors.

Acknowledgments

This work was supported by the intramural research program at the U.S. National Library of Medicine, National Institutes of Health.

References

Gabor Angeli, Christopher D Manning, and Daniel Jurafsky. 2012. Parsing time: Learning to interpret time expressions. In *Proceedings of the 2012 Conference of the North American Chapter of the ACL: Human Language Technologies*, pages 446–455.

Alan R Aronson and François-Michel Lang. 2010. An overview of metamap: historical perspective and recent advances. *JAMIA*, 17(3):229–236.

Eva Banik, Claire Gardent, and Eric Kow. 2013. The kbgen challenge. In *the 14th European Workshop on Natural Language Generation (ENLG)*, pages 94–97.

Anja Belz, Michael White, Dominic Espinosa, Eric Kow, Deirdre Hogan, and Amanda Stent. 2011. The first surface realisation shared task: Overview and evaluation results. In *13th European workshop on natural language generation*, pages 217–226.

Anja Belz. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Nat. Language Engineering*, 14(04):431–455.

Vinay K Chaudhri, Michael A Wessel, and Stijn Heymans. 2013. Kb bio 101: A challenge for tptp first-order reasoners. In *CADE-24 Workshop on Knowledge Intensive Automated Reasoning*.

David L Chen and Raymond J Mooney. 2008. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th ICML conference*, pages 128–135. ACM.

Deborah A Dahl, Madeleine Bates, Michael Brown, William Fisher, Kate Hunicke-Smith, David Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the atis task: The atis-3 corpus. In *Proceedings of the workshop on Human Language Technology*, pages 43–48. Association for Computational Linguistics.

R. Lebret, D. Grangier, and M. Auli. 2016. Generating Text from Structured Data with Application to the Biography Domain. *ArXiv e-prints*, March.

Percy Liang, Michael I Jordan, and Dan Klein. 2009. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 91–99.

Donald A Lindberg, Betsy L Humphreys, and Alexa T McCray. 1993. The unified medical language system. *Methods of information in medicine*, 32(4):281–291.

François Mairesse, Milica Gašić, Filip Jurčíček, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. 2010. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the ACL*, pages 1552–1561. Association for Computational Linguistics.

Yassine Mrabet, Claire Gardent, Muriel Foulonneau, Elena Simperl, and Eric Ras. 2015. Towards knowledge-driven annotation. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 2425–2431.

Thomas C Rindflesch and Marcelo Fiszman. 2003. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *Journal of biomedical informatics*, 36(6):462–477.

Somayajulu Sripada, Ehud Reiter, Jim Hunter, and Jin Yu. 2002. Sumtime-meteo: Parallel corpus of naturally occurring forecast texts and weather data. *Computing Science Department, University of Aberdeen, Aberdeen, Scotland, Tech. Rep. AUCS/TR0201*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.