Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022)

Data Augmentation for Intent Classification of German Conversational Agents in the Finance Domain

Sophie Rentschler, Martin Riedl, Christian Stab, Martin Rückert
Diamant Software GmbH, KI Kompetenzzentrum
Robert-Bosch-Str. 7, 64293 Darmstadt
{s.rentschler,m.riedl,c.stab,m.rueckert}@diamant-software.de

Abstract

In this paper, we focus on improving the intent recognition for a conversational agent. For languages other than English, labeled data needed for training is often limited. Limitations rise even more when moving to specific domains. Here, our goal is to improve the intent recognition for a German conversational agent deployed in the financial sector. We treat this problem as a classification task. Using several augmentation techniques we expand the seed data used for training and compare the performance of the intent classifier. Applying a backtranslation approach using a commercial Machine Translation (MT) engine yields significant improvement ($p < 0.01$) over a baseline system.

1 Introduction

Conversational agents are becoming ubiquitous as lots of companies employ such agents for supporting and extending their services. Based on the applied domain, their languages – specifically, their vocabulary – constantly expand depending on the range of their services as well as the domain they are applied in. Machine learning methods are mainly used to teach conversational agents to react to user requests, called *intents*. For recognizing the intent, usually a *natural language understanding* (NLU) component is used.

In order to train the NLU, for each intent various user utterances are required to understand the user and to discriminate between different intents. Due to the efforts required to manually create sufficient amounts of training data, we investigate if augmentation methods for enriching the training data helps to improve the performance.

In this paper, we tackle various research questions: Is it beneficial to add noise to the data by randomly replacing words or do we really need to have ”human”-readable paraphrases? Also, we will investigate which methods are suitable for automatic paraphrase generation for intents. Most of the previous paraphrasing approaches for dialogue agents focus on training data from the open domain (e.g. booking a hotel, booking a table in a restaurant, calling the police) written in English (Kumar et al., 2019; Quan and Xiong, 2019). In this paper, we research the applicability of augmentation approaches for German for the finance domain.

We present results for a manually created dataset for the finance domain. Using paraphrasing methods to augment training data used for machine learning differs from the typical paraphrasing scenario. Whereas for e.g. text simplification the goal is to generate sentences that can be read by humans, here our goal is to teach the machine learning method to be more robust against textual variations when understanding natural language.
In order to extend the data we use methods based on lexical resources (PPDB (Ganitkevitch et al., 2013), GermaNet (Hamp and Feldweg, 1997)), embeddings and contextual embeddings (BERT (Devlin et al., 2019)) as well as backtranslation using an out-of-the-box machine translation (MT) system. Based on our experiments we achieve significant improvements using backtranslation.

2 Related Work

In recent years, deep learning techniques have become popular to tackle intent classification (Mesnil et al., 2013). This line of work has been continued by combining different tasks of the NLU component into one model (Goo et al., 2018; Haihong et al., 2019). Sequence-to-sequence models have been leveraged to bootstrap intent classification in new features (Jolly et al., 2020). Yet, sufficiently large training datasets are required for such approaches.

Several proposals have been made to resolve the lack of training data for this task and avoid costly generation of suitable datasets by hand. Machine translation (MT) can be used if seed data already exists (Gaspers et al., 2018). Furthermore, exploiting backtranslation techniques, commonly used in MT to overcome shortage of parallel data has become popular for automatic paraphrase generation (Mallinson et al., 2017). Using similar languages for back and forth translation has been proven useful for MT (Hajic, 2000). Whereas backtranslation originates from MT (Sennrich et al., 2016), it recently has been applied to augment data for other tasks such as hate speech detection and transfer learning (Beddiar et al., 2021; Subedi et al., 2021).

Machine Learning tasks are fairly robust to noise in text as long as the corpus is large. Agarwal et al. (2007) report only slight degradation of the system when adding 70% of noise to the text. When adding 40% of noise to the text the system almost performs on par with its competitor which was trained on clean text. Word order and syntactic information are elements which have proven to be mostly irrelevant for text classification[1]. Random word swaps and deletions which first and foremost harm syntax even prove to be helpful data augmentation techniques (Wei and Zou, 2019).

Following the pattern of paraphrase generation, external linguistic resources such as PPDB (Ganitkevitch and Callison-Burch, 2014) or WordNet (Miller, 1995) have been used for retrieval-based approaches (Zukerman and Raskutti, 2002; Babkin et al., 2017; Alva-Manchego et al., 2020). Zhang et al. (2017) established a sentence paraphrasing framework formulated as an encoder-decoder problem. In more recent years, contextualized embeddings were introduced and became the center of attention. BERT (Devlin et al., 2019), ELMo (Peters et al., 2018) and GPT-2 (Radford et al., 2019) have not only been used for paraphrase generation but also for paraphrase candidate ranking (Zhou et al., 2019).

3 Data Augmentation Methods

We use the Rasa framework (Bocklisch et al., 2017) to setup a task-oriented conversational agent. It structures dialogues into two components, namely *Core* and *Natural Language Understanding* (NLU). The Core component takes care of the dialogue management whereas the NLU component performs the entire processing of the text, e.g. tokenization, identification of entities, dependency parsing and classification of intent types. Here, we focus on a basic NLU pipeline including tokenization, intent classification and entity recognition. We aim to improve the task of intent classification by enhancing our training data using augmentation techniques for this task.

Here, we present the augmentation methods we apply in order to enhance the data used to train an intent classifier.

For the resource- and embedding-based approaches we paraphrase one word per intent phrase. We mask words which convey unique information in order to ensure domain-specific words are excluded from paraphrasing. Furthermore, we restrict paraphrasing to words belonging to the categories *verb* and *adverb* for these methods so crucial words remain unchanged. Results (translated to English) of the augmentation methods can be found in Table 1.

PPDB: The multilingual PPDB (Ganitkevitch and Callison-Burch, 2014) is a resource built on bilingual parallel corpora aimed to capture paraphrases. We use the German part of the PPDB for replacing single tokens using the n best-scored words.

GermaNet: GermaNet (Hamp and Feldweg, 1997) is a manually crafted resource. Here, we replace a word by all other words in the same synset.

Embeddings: We consider skip-gram word2vec embeddings (Mikolov et al., 2013)[3]. We paraphrase lexemes' vocabularies using the n most similar words based on the cosine similarity. For this, we sort the vocabulary by cosine similarity and select the n most similar words in order to paraphrase intent samples.

Contextual Embeddings: BERT-based embeddings (Qiang et al., 2020) are used by feeding the intent phrase to the contextual embedding while masking the target word which we want to paraphrase. For replacing verbs and adverbs we proceed in the same manner as with the embeddings approach.

Machine Translation: We make use of the machine translation technique commonly used to over-

[1] We are aware that some machine learning methods are relying more on word ordering than others (e.g. sequence models like CRF or HMM), however, we assume that correctness is more relevant when generating text for humans rather than for machines.

[2] Since the PPDB does not only store lemmatized word forms or infinitives and the pivoting approach uses English as a reference language which is morphologically less complex than German, it groups morphological inflections into the same paraphrase cluster. This is the reason why we find morphological variations of the same verbs used as paraphrases.

[3] We use spaCy vectors which are part of the *de_core_news_md* model containing 276,087 words with vectors and 20,000 unique vectors trained on Wikipedia and OSCAR Common Crawl (Ortiz Suárez et al., 2019)

Augmentation method	Original phrase	Augmented phrases
GermaNet	Show the name of the company.	**Display** the name of the company. **Indicate** the name of the company. **Express** the name of the company.
PPDB	Show the name of the company.	**Shows** the name of the company.[2]
Embedding	Show the name of the company.	**Theatre** the name of the company. **View** the name of the company. **Spectacle** the name of the company.
BERT	Show the name of the company.	**Display** the name of the company. **Demonstrate** the name of the company. **Present** the name of the company.
Machine Translation	Show the name of the company.	**Give me** the name of the company. **Say** the name of the company. **Present** the **company's name**.

Table 1: Paraphrase examples. Underlined target words in the original phrase are replaced by the bold words in the augmented phrases.

come shortage of parallel data. Applying backtranslation, we first translate an intente phrase from a source language (i.e. German) into different target languages and then translate it back into the source language. Here, we use Google's commercial Cloud Translation API[4].

4 Evaluation

Baselines: To judge the performance of the paraphrasing methods we consider three baselines. Gold: The first baseline is represented by the performance without using any augmented data and solely train on the labeled training data.
Random: For the random baseline we replace verbs and adverbs with random words selected from the vocabular of the embeddings. For each training instance we replace one word at maximum.
Duplicate: For the duplicate baseline we add each utterance twice to the gold standard data. This baseline determines whether plainly adding data improves the classifier or more diverse data is needed to improve the system.

Dataset: We evaluate the methods on a manually created German finance dataset for the accounting domain. For the creation of the dataset several people wrote down utterances they would use in a given setting to retrieve information from the dialogue assistant. The dataset comprises 20 intents out of which 12 are exclusive to the finance domain. The remaining eight intents provide domain-independent dialogue elements such as greetings, continuation and abortion of dialogues or confirmation and rejection in selection processes. This data is not balanced across intents. On average, intents are represented by about 44 intent phrases. Examples (translated to English) are listed in Table 2.

Intent	Phrases
who	Who are you? Are you a bot? What's your task?
kpi-help	What KPIs do you know? Which KPIs can you report on? For which KPIs do you have information?
company-set	Let's continue with company XYZ. Change to company XYZ. Please proceed with company XYZ.

Table 2: Baseline dataset: intent phrase examples.

Experimental Setup. Our experiments are based on the Rasa framework[5] from which we use the DIET classifier (Bunk et al., 2020) to train an intent classifier. In this paper, we solely focus on the intent classification and disregard the entity recognition. We randomly split the training data into train, dev and test sets in the ratio of 80/10/10. As we observe high fluctuation in performance between data splittings, for each experiment we use 10 different random seeds to split the data in order to account for outliers which are caused by inconvenient data splittings (Søgaard et al., 2021). In the

[4] https://cloud.google.com/translate

[5] https://rasa.com/

Intents	Gold baseline	Random baseline	Duplicate baseline	BERT	PPDB	GermaNet	Embedding	Top 3 translations NL + IT + FR
affirm	0.6153	0.0047	0.0927	-0.0601	0.0673	-0.0638	0.0402	0.0617
answer-date	0.9153	0.0551	0.0396	-0.0152	0.0469	0.0416	-0.0036	0.0665
answer-taxonomy	0.8222	-0.0440	-0.1451	-0.1166	-0.0773	-0.0097	-0.0504	-0.0707
cancel	0.5503	-0.1979	-0.0044	-0.0540	0.0548	-0.0716	0.0042	0.1521
company-ask-for	0.8951	0.0215	-0.0054	0.0117	0.0326	0.0315	0.0223	0.0470
company-set	0.9452	-0.0406	-0.0212	-0.0167	-0.0110	0.0038	-0.0095	0.0060
compare-kpis	0.9626	-0.0292	-0.0023	-0.0353	0.0087	0.0130	-0.0184	0.0297
customer-overview	0.9382	-0.0340	-0.0131	-0.0009	0.0116	-0.0044	-0.0046	0.0002
greet	0.9139	-0.0231	-0.0678	-0.0012	0.0107	-0.0664	0.0093	-0.0066
kpi	0.9692	-0.0173	-0.0185	-0.0232	0.0000	0.0028	-0.0051	0.0011
kpi-help	0.9344	-0.0179	-0.0018	0.0161	0.0043	0.0268	0.0148	0.0310
op-note-get	0.9001	-0.0440	-0.0446	-0.0420	-0.0069	-0.0203	-0.0386	0.0042
op-note-set	0.8980	-0.0688	-0.0279	-0.0546	-0.0030	-0.0194	-0.0188	0.0203
out-of-scope	0.8676	-0.0169	0.0037	-0.0073	0.0178	-0.0077	0.0008	0.0078
query-op-all-customers	0.9517	-0.0453	-0.0222	-0.0148	-0.0082	-0.0202	-0.0090	-0.0077
query-op-single-customer	0.9524	-0.0708	-0.0300	-0.0102	-0.0048	-0.0148	-0.0240	0.0000
reject	0.5105	-0.1650	-0.0500	-0.1095	0.0879	0.0534	0.0543	0.0895
tell-a-joke	0.9333	-0.0143	-0.0082	-0.0970	-0.0454	-0.0870	-0.0187	0.0667
thx	0.7719	-0.0278	-0.2228	-0.0695	-0.0195	-0.0824	-0.1548	0.0305
who	0.4941	0.0363	0.1224	0.0505	0.1759	0.1150	0.0445	0.1891
macro avg	**0.8371**	**-0.0370**	**-0.0213**	**-0.0325**	**0.0171**	**-0.0090**	**-0.0082**	**0.0359**

Table 3: Report of the F1 scores of the intent classification for the accounting datatset for all paraphrasing approaches.

following, we report scores averaged across these 10 data splittings.

5 Results

Our results for the accounting dataset are reported in Table 3. We show the macro F1 score for the gold baseline and present the delta scores between the augmentation methods and the gold baseline.

The random and duplicate baselines perform inferior to the gold baseline whereas the random baseline works slightly better than the duplicate baseline. We find these differences to be significant[6]. This confirms that the system does not benefit from neither adding pure noise to the training data nor adding data which does not enhance variance in phrasing the same content and benefits overfitting to the training data.

This is in line with the finding that quantity does not beat quality: Augmentation approaches generating the most data (random baseline (+342 intent phrases) and embedding-based approach (+283 intent phrases) vs. BERT (+121 intent phrases) and top 3 translations (+129 intent phrases)) do not necessarily perform best. Indeed, all of these approaches perform inferior to the baseline.

Overall, we observe that using PPDB for augmenting improves the system and we achieve significant improvements with the backtranslation approach ($p = 0.006$). For this approach we tested seven different target languages to extract paraphrases for the source sentence (see results in Table 4). In order to investigate whether the system benefits from an even larger data we combined the backtranslations from different languages. Indeed, the system performs best when combining backtranslations from the top three performing languages (Dutch, Italian and French). However, the improvements are only marginal in comparison to using solely augmentations based on the Dutch translation (0.8715 vs 0.8730).

Whereas we only show the average across ten different data splittings in Table 3 we observe considerable fluctuation in performance across data splittings for a specific group of intents: Both intents which are represented by only a few samples in the training set and intents which tend to have a fixed list of expression (e.g. greet, cancel, reject, thanks, affirm) seem highly susceptible to the random seed used when splitting the data (e.g. the gold baseline F1 score for intent *reject* ranges from 0.0 to 0.89 depending on the data splitting). Here, data augmentation does not eliminate this phenomenon and the splitting of the keywords is mainly responsible for the performance. In contrast, largest performance boost using augmentation methods are on average achieved for these intents (see intents *who* and *reject*), yet dependence on the data splitting

[6] p=0.04 for random baseline and p=0.008 for duplicate baseline using the Wilcoxon signed-rank test.

remains. This suggests that (1) as long as important keywords are present in the given data splitting augmentation methods are specifically beneficial for these intents and that (2) the methods presented cannot make up for missing keywords.

Unexpectedly, the BERT-based approach works worst among all other augmentation methods while its macro average is comparable to the random baseline. In particular, intents *reject* and *answer-taxonomy* suffer from this approach. e.g. the intent *answer-taxonomy* is mostly misclassified as intent *kpi*.

Pivot system	**Macro average F1**	**Increase over gold baseline**
English	0.8572	2.40%
Spanish	0.8468	1.16%
French	0.8630	3.10%
Italian	0.8611	2.87%
Hindi	0.8442	0.85%
Chinese	0.8441	0.84%
Dutch	0.8715	4.11%
All combined	0.8428	0.68%
Top 3	**0.8730**	**4.29%**

Table 4: Results for all languages tested with the MT approach.

Overall, the backtranslation approach outperforms the gold baseline and all other augmentation approaches. However, it is striking that macro averages drop considerably for Chinese and Hindi compared to the rest of the languages. Specifically, for intents *query-op-all-customers* and *query-op-single-customer* the performance drops significantly compared to the baselines. This drop is interlinked as *query-op-all-customers* is misclassified as *query-op-single-customer* and vice versa. Here, again, the intents are very similar and the augmentation does not help the classifier to discriminate the intents. This pattern resembles the behaviour described above: data representing these intents are similar. Paraphrasing this data leads to an overlap causing confusion between the two intents.

The best scores are achieved when combining the outcomes of the three best backtranslation systems. We observe that *answer-taxonomy* is an outlier for this approach as performance decreases by about seven percentage points. Again, this intent is mostly confused with intent *kpi*. However, without exception this intent gets inferior with any of the paraphrasing methods. As expected for the MT approach, the more similar the target language is to the source language (here, German) the more suitable the emerging paraphrases are and thus, the more the classifier benefits from them. This seems apparent comparing Chinese or Hindi scores with the Dutch scores.

6 Conclusion

In this paper, we present several augmentation methods to extend our training data to train a classifier for intent classification. Our best methods achieve significant improvements for the classification task while being easy to implement and not requiring lots of computational resources. We mainly face two limitations regarding the proposed approaches: (1) When we try to build up on lacking data (e.g. missing key words in the original dataset) our methods fail to fill this gap. (2) In case intents are very similar, augmentation approaches seem to rather confuse the classifier than enhance differences which leads to miss-classifications.

References

Sumeet Agarwal, Shantanu Godbole, Diwakar Punjani, and Shourya Roy. 2007. How much noise is too much: A study in automatic text classification. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 3–12. IEEE.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics*, pages 135–187.

Petr Babkin, Md Faisal Mahbub Chowdhury, Alfio Gliozzo, Martin Hirzel, and Avraham Shinnar. 2017. Bootstrapping chatbots for novel domains. In *Workshop at Neural Information Processing Systems on Learning with Limited Labeled Data*.

Djamila Romaissa Beddiar, Md Saroar Jahan, and Mourad Oussalah. 2021. Data expansion using back translation and paraphrasing for hate speech detection. *Online Social Networks and Media*, 24:100153.

Tom Bocklisch, Joey Faulkner, Nick Pawlowski, and Alan Nichol. 2017. Rasa: Open source language understanding and dialogue management. abs/1712.05181.

Tanja Bunk, Daksh Varshneya, Vladimir Vlasov, and Alan Nichol. 2020. Diet: Lightweight language understanding for dialogue systems. *arXiv preprint arXiv:2004.09936*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Juri Ganitkevitch and Chris Callison-Burch. 2014. The multilingual paraphrase database. In *Language Resources and Evaluation Conference*, pages 4276–4283.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764.

Judith Gaspers, Penny Karanasou, and Rajen Chatterjee. 2018. Selecting machine-translated data for quick bootstrapping of a natural language understanding system. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 137–144.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

E Haihong, Peiqing Niu, Zhongfu Chen, and Meina Song. 2019. A novel bi-directional interrelated model for joint intent detection and slot filling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5467–5471.

Jan Hajic. 2000. Machine translation of very close languages. In *Sixth Applied Natural Language Processing Conference*, pages 7–12.

Birgit Hamp and Helmut Feldweg. 1997. Germanet-a lexical-semantic net for german. In *Automatic information extraction and building of lexical semantic resources for NLP applications*.

Shailza Jolly, Tobias Falke, Caglar Tirkaz, and Daniil Sorokin. 2020. Data-efficient paraphrase generation to bootstrap intent classification and slot labeling for new features in task-oriented dialog systems. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*, pages 10–20.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3609–3619.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2017. Paraphrasing revisited with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 881–893.

Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Interspeech*, pages 3771–3775.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, pages 39–41.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora, pages 9–16.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. 2020. Lexical simplification with pretrained encoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8649–8656.

Jun Quan and Deyi Xiong. 2019. Effective data augmentation approaches to end-to-end task-oriented dialogue. In *2019 International Conference on Asian Language Processing*, pages 47–52.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. 2021. We need to talk about random splits. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832.

Ishan Mani Subedi, Maninder Singh, Vijayalakshmi Ramasamy, and Gursimran Singh Walia. 2021. Application of back-translation: a transfer learning approach to identify ambiguous software requirements. In *Proceedings of the 2021 ACM Southeast Conference*, pages 130–137.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Chi Zhang, Shagan Sah, Thang Nguyen, Dheeraj Peri, Alexander Loui, Carl Salvaggio, and Raymond Ptucha. 2017. Semantic sentence embeddings for paraphrasing and text summarization. In *2017 IEEE Global Conference on Signal and Information Processing*, pages 705–709.

Wangchunshu Zhou, Tao Ge, Ke Xu, Furu Wei, and Ming Zhou. 2019. Bert-based lexical substitution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3368–3373.

Ingrid Zukerman and Bhavani Raskutti. 2002. Lexical query paraphrasing for document retrieval. In *Proceedings of the 19th International Conference on Computational Linguistics*, page 1–7.