University *of Ljubljana*
Faculty *of Computer and Information Science*

Automatic Language Translation

Miha Petrišič, Veno Gaube, Armin Komić, Jasna Cindrič, Rebeka Zajc, Gaja Černe, Eva Bolta and Vesna Železnik

Abstract

With the abundance of textual data being available, the need for automatic translators has become necessary more then ever. We have created a machine translator capable of English to Slovene translation. The model was trained to translate primarily military related texts. Domain model was able to achieve BLEU score of 0.32, CHRF 0.54, GLEU 0.34, METEOR 0.50, NIST 6.23 and WER 0.59.

Keywords
machine translation, military corpus, ...

Advisors: Slavko Žitnik

Introduction

In the report we will present our work on the automatic language translation made in part of our Natural language processing course.

We built an English to Slovene machine translator with the help of OpenNMT library which was trained on various corpora consisting of almost 24 million sentences and was then made into a domain specific model by training it on our own data gathered from different military related documents.

Throughout the report we will describe the data used, the process of building a translator and will show some of the results that our model was able to achieve.

Data

In choosing the appropriate corpus for the basis of our machine translator, we opted for the military corpus because it is a very specific topic and largely unexplored topic. Even though our mentors provided us with an extensive military corpus that contained a lot of data, more was needed to successfully build a specialised machine translator, therefore we searched for other bilingual documents. These were obtained mainly from the online sources of the Government and the Ministry of Defence of the Republic of Slovenia and the Slovenian Army. Some examples are the Defence White Book, the Defence Act and various agreements, protocols and laws. We aligned the bilingual documents in SDL Trados Studio, and converted them to an appropriate format (.tmx or .csv) that could be used for the purpose of machine translation training. We also included a large English-Slovenian military termbase that had to be created without the use of CAT tools. This way, we obtained a sufficient amount of data to build and train the machine translator. For the evaluation, we furthermore searched for various other military texts beside the original military corpus which was built for the training and whose part was later used in the evaluation. The newly obtained texts were about military history, military personalities, various military events, international agreements, bilateral relations, NATO, defence strategies etc.

Methods

Setting up the Environment

After having some issues with the SLING environment we switched to Google Colaboratory for running our training process. Although Google's environment has some drawbacks, for example the reserved graphics card capabilities may only last a couple of hours before allowing you to use them again, it has some major positives as well. Although, the issue of limited run time was easily bypassed by saving the model after a certain amount of steps.

One of the major benefits of this environment is that it is extremely easy to use and can be accessed by everyone just by opening the notebook in your browser. This makes it convenient for other people to use, try to reproduce results or construct their machine translator by just changing the data and model settings.

Translation Framework

OpenNMT [1, 2] is a multi-year open-source ecosystem for neural machine translation (NMT) and natural language gen-

eration (NLG). The toolkit consists of multiple projects to cover the complete machine learning workflow: from data preparation to inference acceleration. The systems prioritize efficiency, modularity, and extensibility with the goal of supporting research into model architectures, feature representations, and source modalities, while maintaining API stability and competitive performance for production usages. OpenNMT has been used in several production MT systems and cited in more than 700 research papers. It supports a wide range of model architectures and training procedures for neural machine translation as well as related tasks such as natural language generation and language modeling. OpenNMT [1] provides implementations in 2 popular deep learning frameworks OpenNMT-py, which is more user-friendly multimodal, benefiting from PyTorch ease of use and the second one is OpenNMT-tf which is modular and stable, powered by the TensorFlow ecosystem. Each implementation has its own set of unique features but shares similar goals:

- Highly configurable model architectures and training procedures.
- Efficient model serving capabilities for use in real world applications.
- Extensions to allow other tasks such as text generation, tagging, summarization, image to text, and speech to text.

We decided to go with the OpenNMT-py [2] implementation since we were more familiar with that than the TensorFlow ecosystem. When training GPU-based NMT models, memory size restrictions are the most common limiter of batch size, and thus directly impacting training time. OpenNMT has an external memory sharing system implemented that aggressively shares the internal buffers between clones. In the OpenNMT-py version it is implemented for both data loading to enable training on extremely large datasets that cannot fit into memory, and for back-propagation to reduce memory footprints during training. OpenNMT [2] additionally supports multi-GPU training using data parallelism. Each GPU has a replica of the master parameters and processes independent batches during training phase. Two modes are available: synchronous and asynchronous training. The solution also explicitly separates training, optimization and different components of the model. Each module in the library is highly customizable and configurable with multiple ready-for-use features. The model supports different input types including models with discrete features models with non-sequential input such as tables, continuous data such as speech signals, and multi-dimensional data such as images. OpenNMT [2] packages several additional tools, including reversible tokenizer, which can also perform Byte Pair Encoding (BPE), loading and exporting word embeddings, translation server which enables showcase results remotely and visualization tools for debugging or understanding, such as beam search visualization and profiler.

Preprocessing Data

After all the corpora was gathered we had to process it so that everything had an uniform form. We created scripts that mostly dealt with the *.xml* and *.tmx* formats, transforming them to aligned sentences format. This produces two text files, one with sentences in English and one in Slovene. Everything was also put in lower case for and any type of punctuation marks were separated by spaces. By doing this we got a uniform form that would make it easier to tokenize data when building the vocabulary.

Our general corpora consisted of about 24 million sentences and domain corpora had about 5 thousand of them. Since we have a lot of data for our general model we generously split the corpora to 99% for training and other 1% for the validation and testing set. Since 0.5% of the corpora already means about 120 thousand sentences, we thought that would be enough for evaluating our translations. It is also worth pointing out that we randomized the data division. That was necessary since the whole corpora consisted of different texts and we wanted all of the sets to include approximately the same proportions of those texts.

For the domain corpora we split it into 3 thousand for training and about 1 thousand for testing and validation sets both. The reasoning behind this division is a bit different then with the general corpora. We first wanted to have at least 1 thousand sentences for evaluation. Since this leaves only 4 thousand sentences left we decided to go with a bit larger (at least compared proportionately to the general corpora split) validation set. We did that so we can more easily monitor the learning process since with smaller training set the model might be quicker to over-fit the data.

Building a Vocabulary

We used OpenNMT scripts for building the vocabulary. It does so by inputting training files and building a simple text file with one token per line. The tokens in the file are also sorted based on the frequency that they appear in the texts.

In our case, we just had to input our general training corpora and our domain training corpora. From all the texts we had gathered about 700 thousand tokens from the source texts and more than a million tokens from the target texts were found.

Training the Model

At first we built a general model which was trained on the general corpora consisting of OpenSubtitles 2018, EUparl, EMEA, DGT and ELRC corpora. Besides the training set, validation set was needed for checking meanwhile training, weather our model is over-fitting the training data.

Since settings of our model are the most important factor for building a good translator, there was quite a lot of thought put into it. Here is the thought process behind some of the parameters.

When deciding the amount of training steps model will make, we decided with 10 thousand steps. The training accuracy was of course increasing the more steps we had, but

we were also paying attention to the validation set accuracy. It seemed like the validation accuracy peaked at about 10 thousand steps, that's why it seemed reasonable to stop the model there to prevent over-fitting.

We used the same logic when building the domain model. For it we only used 2 thousand steps, since the training set was quite smaller.

We used recurrent neural networks (RNN) for the encoder and decoder types. Generally transformers perform better, but the implementation in OpenNMT is still under progress and we had some issues with it when trying it out.

When picking the gate types of the RNN we chose Long-short term memory. We also tried Gated recurrent units, but they seemed unable to learn from the training.

Some of the other setting were: 2 encoder and decoder layers, 500 hidden states for encoder and decoder RNN, dropout of 0.3, attention dropout of 0.1, learning rate of 1 and stohastic gradient descent as the optimization algorithm.

Translation Evaluation

Automatic Evaluation

For the evaluation of the results we have used multiple algorithms for evaluating the translation quality. To be able to compare results with other groups we have all agreed to use same toolkit for the evaluation, meaning that all groups have the same implementation of metrics.

We have used NLTK (Natural Language Toolkit) which is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and more.

For the purpose of this assignment we have decided to use following metrics:

- **BLEU** (Bilingual Evaluation Understudy) - Standard algorithm for evaluating the machine translations against the human translations. It was one of the first metrics to claim a high correlation with human judgements of quality, and is still one of the most popular automated and inexpensive metrics. The central idea behind BLEU is that "the closer a machine translation is to a professional human translation, the better it is". BLEU's output is a value between 0 and 1, and indicates similarity between candidate and reference texts. Scores are generally calculated for individual sentences, by comparing them with a set of good quality reference translations. To estimate overall quality, scores are then averaged over the whole test dataset.

- **CHRF** (character n-gram F-score) - This metric is based on character n-gram precision and recall enhanced with word n-grams. The tool calculates the F-score averaged on all character and word n-grams, where the default character n-gram order is 6 and word n-gram order is 2. The arithmetic mean is used for n-gram averaging.

- **GLEU** (Google BLEU) - While BLEU adds a brevity penalty and combines the precision for each n-gram level geometrically, GLEU defines a recall measure as the ratio between true positives and total n-gram count in the reference (tpfn), and computes the final score as the minimum between precision and recall. Similarly GLEU outputs a value between 0 and 1, indicating similarity between candidate and reference texts.

- **METEOR** (Metric for Evaluation of Translation with Explicit ORdering) - The metric is based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision. It also has several features that are not found in other metrics, such as stemming and synonymy matching, along with the standard exact word matching. The metric was designed to fix some of the problems found in the BLEU metric, and also produce good correlation with human judgement at the sentence or segment level. This differs from the BLEU metric in that BLEU seeks correlation at the corpus level.

- **NIST** (name comes from the US National Institute of Standards and Technology) - It is based on the BLEU metric, but with some alterations. Where BLEU simply calculates n-gram precision adding equal weight to each one, NIST also calculates how informative a particular n-gram is. That is to say when a correct n-gram is found, the rarer that n-gram is, the more weight it will be given. NIST also differs from BLEU in its calculation of the brevity penalty insofar as small variations in translation length do not impact the overall score as much.

- **RIBES** (Rank-based Intuitive Bilingual Evaluation Score) - The focus of the RIBES metric is word order. It uses rank correlation coefficients based on word order to compare SMT and reference translations. The primary rank correlation coefficients used are Spearman's ρ, which measures the distance of differences in rank, and Kendall's τ, which measures the direction of differences in rank. RIBES works well for language pairs having very different grammar and word order.

Adequacy Fluency

The Adequacy Fluency evaluation template was used to assess the quality of the machine translation. We worked with the source and reference translation, as well the machine translation output. 150 machine-translated sentences that accounted for approximately 2,800 words in total were randomly selected for evaluation. Three different evaluators were then assigned to assess 50 sentences each.

The following metrics were used for the evaluation:

- **Fluency**: grammatical correctness, idiomatic word choices

- **Adequacy**: the meaning of the translation in comparison to the source

To evaluate fluency, sentences were assessed and graded based on the following scale:

To what extent is the translation well-formed grammatically, contains correct spellings, adheres to common use of terms, titles and names, is intuitively acceptable and can be sensibly interpreted by a native speaker?

- 4 Flawless
- 3 Good
- 2 Disfluent
- 1 Incomprehensible

To evaluate adequacy, sentences were assessed and graded based on the following scale:

How much of the meaning in the reference translation is also expressed in the translation?

- 4 Everything
- 3 Most
- 2 Little
- 1 None

Once the average scores of both fluency and adequacy were calculated, the evaluators compared their judgements and measured the level of their agreement to determine how consistent they were with their evaluation.

Results

After training our models we were interested in their performance. Having general and domain models trained, we wanted to see whether domain model will outperform general model, when translating domain specific texts. To show performance of each model, and to be sure that the results are not impartial we used multiple aforementioned automatic translation evaluation metrics and also some non-automatic quality evaluations. Having problems implementing RIBES metric, we have replaced it with WER. Because some of the methods are case-sensitive, we transformed all texts to lower case before doing any evaluations.

As we started with training general model first we need to first evaluate its performance. This gives us a reference point for evaluating improved versions of our model and ability to compare our domain model results with it. Table 1 shows multiple metric scores of our general model being tested on general text. For a comparison, we also added evaluation of google translated texts to the mentioned table. As we expected google translator of course performs much better than our model, but our goal was not to build generally better translation tool but to to achieve better results on domain specific texts.

	General model	Google translator
BLEU	0.0725	0.1689
GLEU	0.1123	0.2082
CHRF	0.2209	0.5549
METEOR	0.2078	0.3411
NIST	2.6406	5.4562
WER	0.90364	0.6826

Table 1. Performance of general model compared to Google translator on general texts.

	General model	Domain model
BLEU	0.0374	0.3230
GLEU	0.0844	0.3392
CHRF	0.2708	0.5429
METEOR	0.1956	0.5003
NIST	2.0913	6.2250
WER	1.0099	0.5935

Table 2. Performance comparison of general and domain models when translating domain specific texts.

Note that; BLEU, GLEU, CHRF, METEOR and NIST metrics have higher value for similar texts and lower for more texts that are different whereas lower WER value represents smaller error in translation.

For the second part of the assignment we have tried to work on our general model and improve it in a way, so it becomes domain specific. For this experiment we used a test set containing texts related to military domain. We wanted to find out, if our domain model performs better than general model. Using both models we generated translations and using aforementioned methods evaluated their performance. Table 2 shows performance comparison of general and domain models when translating domain specific texts. We can see that domain specific model significantly improved translations therefore giving much better scores for each metric.

Compared to Google Translate

Significant improvement in translating military texts gave us idea to compare our translations with the ones generated by Google translator. For this experiment we first prepared a script that uses Google translator API to translate text from english to slovene. To make the comparison more precise we developed a script in a way so it translates each sentence individually. We then took our military test set, translated it and ran evaluation. To see if our translations were better than those produced by Google translator, we put all scores side by side in table 3.

From the above results we can conclude, that our domain model produces slightly better translations than those generated by Google translate. We can interpret this as our model being better or this comparison not being fair, because Google translator is general and not focused to military domain. Ei-

	Domain model	Google translator
BLEU	0.3230	0.1623
GLEU	0.3392	0.2050
CHRF	0.5429	0.5774
METEOR	0.5003	0.3726
NIST	6.2250	5.0464
WER	0.5935	0.7029

Table 3. Performance comparison of our domain model and Google translator when translating domain specific texts.

ther way, we can conclude that our military domain based translator performs well.

Discussion

One of the things that becomes very evident when looking at translations is that there is a lot of unknown words. There can be many reasons for it, for example that the model was not trained long enough or the words in training set differ significantly from those in testing sets. One of the issues that might be more reasonable is related to the environment we used. The vocabulary our model used was not the whole vocabulary used in the training data, but a smaller portion of it (about 200 thousand tokens). This was done because of the memory limitations when working with Google Colaboratory platform.

It is hard to pin down the exact problem, but after trying different methods, including using larger vocabulary and tuning network settings we are fairly sure that our model did not have enough steps to train. We came to this conclusion since we had a similar issue, when we used a different, much smaller training set (EUparl corpus) that was resolved with increasing the number of steps. Although, like we mentioned in the methods part of the paper, we had certain rationalizations for the number of steps that we had used. The further work on this project should definitely include longer training process.

Adequacy & Fluency

1 - VERY GOOD

In changing conditions , taking preventative measures and forestalling the grounds for the origins of conflicts and crises is essential .	v spremenjenih razmerah je bistveno preventivno ukrepanje in preprečevanje vzrokov za nastanek konfliktov in kriz .
In doing so, the solutions must be simple , because the very restricted time for measures does not allow complex solutions .	pri tem morajo biti rešitve enostavne , saj zelo omejen čas za ukrepanje ne dovoljuje zapletenih rešitev .
Along with economic crises , especially in the case of economic pressures and other forms of economic warfare , the Slovenian economy could be confronted with serious difficulties .	ob ekonomskih krizah , predvsem ekonomskih pritiskih in drugih oblikah ekonomskega vojskovanja , bi se slovensko gospodarstvo lahko soočilo z resnimi težavami

Fluency: 4, Adequacy: 4 Apart from the sentence beginning in lowercase, the translation is grammatically correct and the meaning completely adequate to the source.

2 - GOOD

Membership of the EU and NATO are of particular importance to Slovenia as they represent the strategic mechanisms for ensuring a stable and secure environment which will enable the peaceful development of the country	članstvo v eu in nato so bile predvsem v sloveniji , saj jih bo omogočal strateške mehanizme za zagotavljanje stabilnega in krepitev okolja , ki bo omogočala celostno razvoj države .

Fluency: 2, Adequacy: 3 Since the MT is case-insensitive, there are spelling errors; “eu” and “nato” are not capitalized and when reading the isolated sentence “nato” may be read as a connector rather than a name of the organisation. Some wordings are awkward in the translation (*so bile predvsem, zagotavljanje krepitev okolja*) and the grammar is not flawless (subject and the verb do not match: *mehanizme. . . , ki bo omogočala*), but the translation nevertheless expresses much of the meaning in the source text and was rated as an adequate translation.

In cooperation with allies, the SAF will provide the necessary capabilities for effective operation in the air space .	v sodelovanju z zavezniki bo zagotovljeno tudi zmogljivosti za učinkovito delovanje v letalskem operacijah .

Fluency: 2, Adequacy: 3 The sentence in the translation begins with a lowercase letter, which should have been capitalized. The abbreviation SAF does not appear in the translation and therefore passive voice is used. The verb and the adjective do not match with their corresponding plural nouns (*bo zagotovljeno tudi zmogljivosti; v letalskem operacijah*). The meaning was nevertheless expressed in a way that would be understandable to a native Slovenian speaker.

Member of special forces of Territorial Defence of Slovenia during the liberation fight in 1991 .	pripadnik sil za teritorialno obrambo slovenije v ljubljani pa leta 1991 .

Fluency: 2, Adequacy: 3 The first letter of the sentence is not capitalized and some of the meaning from the source text got changed with the translation (instead of "med osvobodilnim bojem" it says "v ljubljani"). Overall the main part of the meaning from the source text was conveyed in the target text and it's still understandable.

3 - AVERAGE

This ordinance reduced the size of defence plans in some of the Ministries and introduced a requirement that defence plans also include crisis management measures .	z omenjeno uredbo se je število obrambnih načrtov obseg obrambnih načrtov povečalo za vrsto obrambnih načrtov tudi za obvladovanje obrambnih načrtov .

Fluency: 2, Adequacy: 2 The translation is difficult to read due to the repetitions of "obrambni načrti" which is twice unreasonably added in the target text. There are several omissions (*crisis, Ministries, requirement*) and mistranslations (*reduced - povečalo*). The style is sometimes awkward (*obvladovanje obrambnih načrtov*). The grammar, however, is good. Was it not for the omissions and the additions, the translation would somewhat still make sense.

Previously the main task of the defence administrations in the area of civil defence was to direct and coordinate the work with the bodies responsible for defence planning	glavna naloga obrambnega uprave na področju civilne obrambe je bil neposredno vplivala na organe , ki so odgovorni za obrambno načrtovanje .

Fluency: 2, Adequacy: 2 Since the MT is case-insensitive, the letter at the beginning of the sentence is not capitalized. An adequate translation for the word *previously* is missing. The adjective *obrambni* is not declined correctly, the verb structure is off and it does not convey the message completely accurately. Despite some awkward grammar, a native speaker would still be able to make some sense of this translation.

There was also a MORiS brigade participating in the exercise tasked with setting up ambushes .	v vaji je sodelovala tudi brigada moris , ki je imela nalogo postavljanja postavljanja .

Fluency: 2, Adequacy: 2 The first letter of the sentence is not capitalized and there is also a repetition of "postavljanja". Because of this a part of the meaning of the source text is lost.

4 - BAD

The living environment is a complex structure that has a decisive impact in ensuring the various aspects of security - global , international , regional , national and even individual - as it determines the possibilities for development and defines relations both in the international community as well as at the regional and national levels .	življenjsko okolje je , ki bo odločilno za zagotovitev različnih vidikov varnosti - globalni , mednarodne ,regionalne , regionalne , regionalne , regionalne , regionalne , regionalne , regionalne , regionalne , regionalne in druge odnose tako v mednarodni skupnosti kot tudi v pokrajinskih državah skupnosti .

Fluency: 1, Adequacy: 2 This example is to show how the same word (*regionalne*) was repeatedly added in the MT and others (*national, individual*) were not translated. While the first part alone would have been rated a 4, is the second part bad, although the grammar is good. There are furthermore omissions (*as it determines the possibilities for development and defines relations*) and mistranslations (*v pokrajinskih državah skupnosti*).

5 - INCOMPREHENSIBLE

The concentration camps were set up in various locations and facilities .	v objektih so bili poslani vojaki tajnosti in prostorov .
Parents from the other republics kept coming to Slovenia to look for and visit their sons .	v ljubljano so se na slovenijo postavili tudi skladišča iz drugih republike slovenije .
In this case Slovenia may be confronted with a problem of refugees and violence spilling over from areas of conflict .	tako bi lahko sloveniji lahko soočilo z rešitvami in albanci na obeh področjih , ki se lahko soočilo .
The advance of the YPA was not a routine movement .	članstvo jla ni šlo za šlo .

Fluency: 1, Adequacy: 1 These are some examples that were completely mistranslated and are incomprehensible. We can recognize only a few words from the source text (*prostorov, slovenije, soočilo*).

Evaluation of Terminology Translation

The evaluation of terminology was conducted on 2,000 words of the translated text, with 104 extracted terms. The translated terms were evaluated on a scale of adequacy. Entirely correct translations were marked as "adequate," translations that were partially correct as "partially adequate," and completely incorrect translations as "inadequate."

Source term	**Translated term**	**Adequacy of translation**
International commitments	Mednarodne zaveze	adequate
International operations	Mednarodne operacije	adequate
Members of the Slovenian Armed Forces	Pripadniki slovenske vojske	adequate
High Performance Computing	Visoko delovati	Inadequate
Capability Development Plan	Kadrovski načrt razvoja	Inadequate
SAF land component	Kopenska vojska	Partially adequate

Table 4. Sample of the evaluation process

The ratio between correctly translated and incorrectly translated terminology was almost even, with 27 of the terms marked as "adequate," 7 of the translated terms marked as "partially adequate," and 19 as "inadequate". The partially adequate translations included terms with incomplete translations, such as "*bela knjiga*" for "Defence White Paper" and "*kopenska vojska*" for "SAF land component" or phrases with one component translated correctly and another incorrectly, such as "*vojaška skupina*" for "battle group" and "*kadrovski viri*" for "human resources." Inadequate translations were most likely a result of partial recognition of the term by the software. This resulted in a translation of the term with a completely different term. For example, "Officer of the Slovenian Armed Forces" was translated as "*republiški štab slovenske vojske,*" "security sector" as "*nacionalni sector,*" "peacetime" as "*izredno in vojno stanje,*" "alliance member states" as "*države članice,*" and "air combat" as "*bojno bojevanje,*" to name a few. Some other examples of incorrect translations are "*teritorialna*" for "legislative," "*sekretariat*" for "executive," and "*mlf" (Multi National Landforce)* for "partner countries." 51 of the extracted terms remained untranslated.

Adequacy of translation	**Number of terms**
Adequate	27
Partially adequate	7
Inadequate	19
Untranslated terms	51
All terms	104

Table 5. Results of terminology evaluation

Conclusion

The quality of the translations varied heavily, ranging from nearly perfect to almost incomprehensible. The evaluators were pretty consistent with their assessment, with the fluency average scores of 2.4, 2.78 and 2.66, and the adequacy average scores of 2.27, 2.9 and 2.74. In general, the translations scored higher in adequacy than in fluency, with the content still rather easily understandable to a Slovenian native speaker, despite several grammatical errors (declension, capitalization) and some awkward wording.

In general the model was very hit-or-miss. We had already explained some possible solution, but let us restate some, for which we believe might increase the quality of the model. We believe that increasing the training time on general corpora might be beneficial. There is a number of military documents that could also be included for building a bigger domain corpora. Switching to a different framework that has transformers fully implemented might also be reasonable. These are some of the main points we derived and believe that have to be looked at if further work on the subject is done.

Acknowledgments

We acknowledge our mentors Slavko Žitnik and Špela Vintar for their input and guidance.

References

[1] Guillaume Klein, Francois Hernandez, Vincent Nguyen, and Jean Senellart. Evolution of Collective Behaviour in an Artificial World Using Linguistic Fuzzy Rule-Based Systems. *AMTA*, 1(1):102–109, 2020.

[2] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, July 2017. Association for Computational Linguistics.