Evaluating the Accuracy of Cloud NLP Services Using Ground-Truth Experiments

Frank Pallas
Information Systems Engineering
TU Berlin
Berlin, Germany
fp@ise.tu-berlin.de

Dimitri Staufer
Information Systems Engineering
TU Berlin
Berlin, Germany
ds@ise.tu-berlin.de

Jörn Kuhlenkamp
Information Systems Engineering
TU Berlin
Berlin, Germany
jk@ise.tu-berlin.de

***Abstract*—Cloud services for natural language processing (NLP) increasingly establish as viable alternatives to self-maintained and self-trained NLP pipelines. In particular, they feature low access barriers and management overhead, a pay-as-you-go pricing model, and elastic scalability allowing to process large amounts of natural language data ad hoc. Any deliberation about employing cloud NLP services in practice does, however, face the challenge that so far, little is known about the accuracy provided by such services as well as about how to conduct respective quality assessments.**

In this paper, we therefore present a method for evaluating the accuracy provided by cloud NLP services and apply it to cloud services for three prominent NLP tasks offered by Amazon, Google, Microsoft, and IBM. Our results show significantly different accuracies as well as different dependencies on the specifics of input data among the covered providers. Our insights therefore allow for a more evidence-based quality-driven choice of the provider to be used for NLP in practice. Furthermore, the general approach employed may also serve as a blueprint for additional future evaluations of cloud NLP services for other tasks or offered by other providers.

***Index Terms*—NLP, natural language processing, experiments, evaluation, cloud computing**

I. Introduction

Natural language processing (NLP) today forms an indispensable basis for many information-heavy products and services, ranging from search engines over chatbots, smart home assistants and social media analysis to automated text summarization and translation. Large as well as smaller players are therefore employing or at least investigating the capabilities of NLP to create novel products and services or to improve existing ones.

Unlocking the full potential of NLP typically requires to laboriously familiarize with its underlying theoretical foundations as well as with respective toolchains, to implement complex NLP pipelines, and to extensively train own models tailored to the respective application domain and the tasks that are to be solved. In many cases, however, the last bit of functional capabilities is not required at all while other properties such as ease of use, low upfront costs, or high scalability allowing to process large amounts of natural language data ad-hoc are of significantly higher relevance.

This is where cloud NLP services come into play. Offered by most major cloud providers as well as by specialized, NLP-focused ones, such services are available for many different tasks existing in the NLP domain. They are typically usable via common web service interfaces or easy-to-use programming libraries, billed on a per-use basis, and – what is particularly relevant for processing large amounts of natural language data – provide virtually unlimited scalability.

Opting for the use of out-of-the-box NLP services, in turn, requires to consciously choose between different providers of such services. Relevant factors to be taken into account in this choice particularly include (i) the accuracy of returned results, (ii) the costs to be expected, and (iii) the performance provided. Depending on the use-case, the relevance of these factors will differ, but for the majority of cases, we assume accuracy to be considered most important, followed by costs and performance. However, little information is available on how accurate existing cloud NLP services are and, noteworthily, on how to conduct a respective comparison in a realistic, fair, and unbiased manner.

In this paper, we therefore present a method for the experiment-driven evaluation of cloud NLP services' accuracy based on real-world ground-truth data and apply this method to cloud services offered by major providers for three prominent NLP tasks. In particular, our contributions include:

- an experiment design for evaluating the accuracy of cloud NLP services based on real-world ground-truth data
- an approach for gathering such ground-truth data that comprises randomized inputs and accurate reference results.
- based on the above, an evaluation and comparison of existing cloud NLP services for sentiment analysis, named-entity recognition, and text classification offered by Amazon, Google, Microsoft, and IBM.

We develop our approaches and insights as follows: Based on some background and related work presented in section II, we describe our experiment design in section III. Ground-truth datasets employed in our experiments are discussed in some more detail in section IV and the specifics of our experiment execution are provided in section V. Our experiment results are presented and discussed in section VI. Section VII concludes.

II. Background & Related Work

NLP is a comparably broad field comprising a variety of lower- and higher-level tasks such as part-of-speech tagging, dependency parsing, text summarization, or machine translation. In our experiments, we focus on the three NLP tasks sentiment analysis, named-entity recognition, and text classification as valuable evaluation targets. These can be briefly described as follows:

- **Sentiment Analysis** refers to the extraction of "peoples opinions, sentiments, evaluations, attitudes, and emotions from written language" [1]. It is of particular relevance for application domains such as the analysis of social network data for marketing or political purposes or the prioritization of customer tickets.
- **Named Entity Recognition** stands for the identification and classification of well-known entities such as persons, locations, or organizations from given texts [2]. Its application domains range from search engines over the automated dispatching of consumer inquiries to text summarization.
- **Text Classification**, in turn, is the automated mapping of given text to a set of (potentially hierarchically structured) categories [3]. Typical applications include the thematic sorting of documents (e.g., to list news items referring to a particular sport) and respective suggestion mechanisms.

As for the accuracy-oriented evaluation of respective cloud NLP services aspired herein, a large body of related research exists on experiment-driven evaluation of cloud services from a cloud user perspective [4], [5]. However, publications mainly focus on evaluating low level infrastructure services provisioning storage [5]–[7] or compute [8] capacity. Such work typically studies traditional service qualities such as performance [5], elasticity [9], or data consistency [4]. Even though respectively established approaches are not suitable for observing the accuracy of NLP services, we use existing best practices in our latency [6] and cost [10] considerations.

For non-cloud NLP, in turn, accuracy-oriented experiments have been widely conducted. Those focusing on sentiment analysis [11], [12] do, however, typically employ a binary measure of correct/incorrect analysis while the average polarity offset (APO) referred to herein (see section IV-B) is more appropriate. For named entity recognition, accuracy evaluations [13]–[15] typically employ established, expert-generated ground-truth datasets which appear to be inappropriate for cloud NLP services (see section IV-A). Existing work on the accuracy evaluation of text classification, in turn, is rather sparse and, where existing, mostly focuses on the selection of and parameter optimization for underlying machine-learning techniques [16], [17].

Most importantly, however, we have to clearly distinguish between the party operating the NLP service (the cloud provider) and the one using it (the client) in the context of cloud NLP services. Of these, existing approaches are rather aligned with the perspective of the cloud provider, focusing on parameter optimization as well as time and cost of model training, assuming a whitebox view on the NLP service. In contrast, our approach is aligned with the perspective of a cloud consumer, with a focus on evaluating accuracy, cost and latency of service usage and assuming a blackbox view on the NLP service. To the best of our knowledge, such a structured, client-perspective evaluation of cloud NLP services has not been conducted before.

III. Experiment Design

Our experiment approach and design rest upon a couple of underlying considerations regarding what is to be tested (goal) for which systems (service under test) against what reference (ground-truth data) through which measurands (observations & metrics). All these aspects shall be briefly elucidated below.

A. Goal

The primary goal of our experiments is to evaluate the accuracy of results provided by different cloud NLP services. We do so following the approach laid out by Resnik and Lin [18], which is based on a ground-truth dataset comprising samples of representative input data and respectively expected outputs (herein called reference results). The accuracy of the system is therefore the level of the "system's agreement with the ground-truth" when confronted with the input data. This leads us to our first research question:

- **Q1**: How good is the accuracy of cloud NLP services?

We assume that the accuracy is dependent on various factors of the input data such as the length of the provided text To the best of our knowledge, we conduct the first experiment-driven evaluation of cloud NLP services and thus, factors with a significant influence are unknown to cloud users and experimenters. Striving to illuminate the impact of such factors, we raise the additional research question:

- **Q2**: What factors influence the accuracy of the different services and how?

Besides our primary goal of assessing accuracy, we are also interested in additional factors relevant for deciding on the usage of a cloud NLP service for solving a given task. These particularly include the costs that are to be borne when using a service at scale as well as the latencies to be expected. In addition to the accuracy-related ones, we therefore also want to answer the following, third research question:

- **Q3**: What are the costs and latencies of cloud NLP services?

We use the term cost to describe monetary platform consumption from a cloud user's perspective. Regarding latency, we are interested in client-side request-response latency.

B. Service Under Test

We focus our experimentation efforts on accessible service offers. Thus, we investigate widely established cloud providers and ignore highly specialized ones. Precisely, we concentrate on NLP services that are part of the cloud platforms AWS, Google Cloud Platform, IBM Cloud, and Microsoft Azure and explicitly neglect offers from, e.g., TextRazor, Aylien, or Dandelion herein. The covered, major cloud providers offer a multitude of different NLP services dedicated to different tasks. We select three NLP tasks (see section II) that are widely supported by each cloud platform.

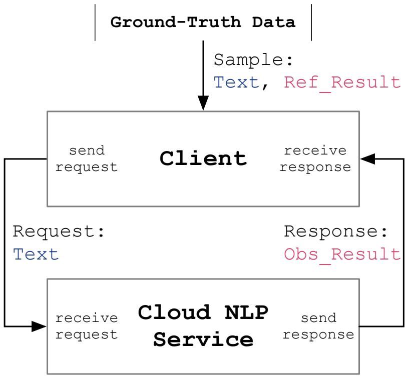

Fig. 1. High-level experiment design.

C. Ground-Truth Data

Each experiment assumes an existing dataset representing ground-truth data. The dataset consists of samples, each of which comprises an input text and a reference result. Of these, the input text serves as input to an invocation of the service under test while the reference result serves as a reference value for analyzing accuracy. In section IV, we give a detailed description of our method for obtaining appropriate ground-truth data.

D. Observations & Metrics

For each service invocation, we observe the start- and end-time as well as the result provided in the response. Based on these observations, we calculate two metrics: As a measure for accuracy, we calculate the distance between the reference result and the observed result using a distance function $d(ref\_result, obs\_result)$. As the function d differs for different NLP tasks and cloud service offers, we provide a detailed description of each distance function in section IV. Client-side request-response latency, in turn, is calculated as the difference between initiating a request and receiving the respective response on the client. Moreover, we also observe the aggregated costs raised by each cloud platform on a per-service level.

IV. Ground-Truth and Quality Metrics

Following the approach for determining the accuracy of results laid out above, one – if not the – core challenge is to identify or generate the ground-truth dataset to evaluate against. Any such ground-truth dataset must fulfill several requirements which we identify first before elaborating on the ultimate datasets we employed in our experiments.

A. Requirements

To actually serve the intended purpose and evaluation approach, a ground-truth dataset must at least fulfill the following requirements:

- **R1 - Bipartite structure:** First and foremost, the ground-truth dataset must comprise bipartite items, consisting of input data and a reference result considered to be the correct output for the input data
- **R2 - Realistic data:** The input data must be a realistic target for the particular service to be evaluated
- **R3 - Reliability of reference results:** The reference results must be considered sufficiently reliable in matters of actually being correct for the associated input
- **R4 - Dataset size and diversity:** The dataset must be sufficiently large and diverse to allow for reliable and generalizable results

Basically, ground-truth datasets meeting these requirements can be created in two different ways: Either, an expert-driven approach can be followed, meaning that skilled experts in the domain of the service to be evaluated manually create a reference dataset. This approach leads to high-quality reference results but at the same time raises significant efforts and therefore becomes increasingly impractical for creating large-scale datasets. Alternatively, an approach of implicit crowdsourcing can be followed where a pre-existing dataset originally created for a different purpose is re-used as ground-truth data. This approach easily provides high-volume datasets but may be less reliable with regard to the quality of reference results. Given a sufficient size, however, this quality detriment can be considered less severe, due to the “wisdom of the crowd” effects [19].

In our particular context of cloud NLP services, an additional constraint applies to the selection or creation of suitable ground-truth datasets: Given that the providers of considered services also train their internal models based on reference data, we must as far as possible avoid using the same datasets in the evaluation to prevent adulterated results emanating from pre-fitted internal models. Even though this risk softens with increasing dataset size and diversity as pre-fitting would then be less punctual and closer to realistic input data, this leads to the following additional requirement:

- **R5 - Low risk of pre-fitted models:** It must be as unlikely as possible that the service is pre-trained with the same dataset used during evaluation. With increasing dataset size and diversity, this requirement can be relaxed gradually

Based on these considerations and in the light of our intention to provide a qualitative evaluation of NLP-related cloud services that is as generalizable as possible, we opted for the approach of implicit crowdsourcing as the basis for our ground-truth dataset. We therefore consciously decided against using well-established reference datasets explicitly dedicated to NLP such as the CoNLL-2003 Corpus[1] employed by [20] or the manually annotated WikiGold Corpus used by [14] for named-entity recognition or the Reuters Corpora[2] for text classification, as 1) we assume a high probability of these datasets to be also used as training data by the considered cloud providers (conflicting with R5) and 2) some of these datasets are of only limited size, circumventing our intention to conduct experiments with explicitly broad coverage (R4).

[1] https://www.clips.uantwerpen.be/conll2003/ner

[2] https://trec.nist.gov/data/reuters/reuters.html

Instead, we decided to use larger-scale datasets for evaluating the selected cloud NLP services against. Once having gathered or created such a dataset that sufficiently meets all requirements mentioned above for each of the three covered NLP tasks, we can then confront services with the respective items and evaluate their accuracy based on appropriate metrics.

After identifying and assessing several pre-existing datasets as well as the possibilities to generate appropriate ground-truth datasets on our own for each of the three NLP tasks, we decided to use a pre-existing large-scale dataset of customer reviews for the task of sentiment analysis and created two own, task-tailored datasets for named-entity recognition and text classification. In the following, we describe these reference datasets as well as the associated accuracy metric(s) we employed and briefly discuss their validity and limitations with regard to the requirements identified above. In some cases, we also gathered additional indicators beyond the primary metrics to investigate the influence of respective factors on the provided accuracy (see research question 2). Where this is the case, these additional indicators are also presented briefly.

For those cases where we generated own datasets, we did so through an approach we call "search-based indirection of reference results". More details on this approach are provided in IV-C.

B. Sentiment Analysis

As ground-truth for the NLP task of sentiment analysis, we use the publicly available "Yelp Open Dataset" of online reviews.[3] The dataset includes subsets of Yelp's businesses and user data, particularly comprising more than 8 million user reviews written over the course of 15 years. Each review consists of (i) the review's text string, (ii) the review's star rating (integer from 1 to 5), (iii) a "usefulness" attribute, which is an integer from 1 to 10 derived from other users' feedback, and finally (iv) the date and time of the review.

By using the reviews as ground-truth for our sentiment analysis evaluation, we assume that a user's star rating correctly reflects the sentiment embodied in the review's text. One star means very negative sentiment, and five stars mean extremely positive sentiment.

Measurand – Absolute Polarity Offset: Our intended measurand for sentiment analysis is the distance between the actual sentiment expressed in the Yelp star rating of a particular review s_{ref} and the recognized one reported for that review by the cloud NLP service $s(o)$. We name this distance the absolute polarity offset (APO $\in [0, 4]$) (1). An APO of 0 represents the highest and a value of 4 the lowest possible accuracy.

$$APO = |s_{ref} - s(o)| \tag{1}$$

As different cloud providers deliver their results in different form, we present three functions (see eq. 2-4) for converting the observed sentiment o to a Yelp star rating $s(o) \in [1, 5]$.

Beginning with IBM's *Watson Natural Language Understanding* (WNLU) service and Microsoft's *Text Analytics* (MTA) service, both return a real number between 0 and 1 with 0 indicating a most negative sentiment, and 1 meaning most positive. Thus, we assume $\{0 \leq o \leq 1 | o \in \mathbb{R}\}$. Therefore, we can describe $s(o)$ as a function of o:

$$s(o) = 4o + 1 \quad | \; WNLU, \; MTA \tag{2}$$

Similarly, Google's *Natural Language* service (GNL) returns one real number between -1 and 1: $\{-1 \leq o \leq 1 | o \in \mathbb{R}\}$. Thus, the conversion is as follows:

$$s(o) = 2o + 3 \quad | \; GNL \tag{3}$$

Amazon's *Comprehend* service (AC) returns four real numbers $o = (o_p, o_n, o_g, o_m)$ for *positive* (o_p), *neutral* (o_n), *negative* (o_g), and *mixed* (o_m) sentiment, ranging from 0 to 1, that add up to 1: $\{0 \leq o_p, o_n, o_g, o_m \leq 1 | o \in \mathbb{R}\}$. For example, "positive: 0.32, neutral: 0.02, mixed: 0.08 negative: 0.58". We convert these four numbers to the Yelp star system as follows:

$$s(o) = 4o_p + 2o_n + 2o_m + 1 \quad | \; AC \tag{4}$$

The most negative sentiment (o_g=1) results in the lowest possible star rating (s=1) because it implies $o_p, o_n, o_m = 0$. This allows us to completely remove o_g from equation 4.

Additional Indicators: To allow for a more detailed analysis of results, we also calculated several additional indicators characterizing the employed text items for the case of sentiment analysis. Basically, sentiments are typically expressed through so-called opinion words (such as "awful" or "great"). However, solely relying on such opinion words is "not sufficient for sentiment analysis" [1]. Analyzing accuracy in the light of such opinion words being present therefore proposes interesting insights: The more accurate a service's APO remains with fewer opinion words, the less is the service reliant on them. As an indicator, we therefore calculated both the number and proportion of adjectives present in a review text using NLTK. Additionally, we also collected the individual reviews' length and usefulness rating, expecting that longer and more useful reviews will – on average – lead to a higher accuracy for the sentiment analysis.

Validity and Limitations: The Yelp datasaet contains a total of 8,021,122 reviews by 1,968,703 users for 209,393 businesses, summing up to 9.8 GB of user-written ground-truth text. Online reviews are a core application area for sentiment analysis and can thus be considered realistic input data (R2). The star-ratings of reviews are the reference results (R1) and can be assumed to be reliable (R3) as users themselves summarize their sentiment in these ratings – the risk of a positive review carrying a low star-rating is thus negligible, especially with a high number of reviews being used. Finally, the Yelp dataset comprises reviews for a high diversity of businesses, including gastronomic ones (e.g., restaurants or cafés), entertainment facilities (e.g., cinemas or night clubs), and public institutions (e.g., libraries or hospitals) and thus covers a broad range of subjects. It therefore provides a sufficiently large and diverse (R4) basis for evaluation.

[3] https://yelp.com/dataset

Limitations exist with regard to the risk of pre-fitted models (R5): The Yelp dataset is well-known and easily available in a processing-friendly form. There is thus a certain probability that cloud NLP providers use it – among others – to train their models for sentiment analysis, ultimately leading to results pre-fitted to this particular dataset. However, the dataset is sufficiently large and diverse to actually make it a valid cross-section of texts relevant for sentiment analysis. A model pre-fitted to this corpus would thus not necessarily invalidate its use. We thus consider the benefits of the Yelp dataset in matters of size, diversity, and reliability of reference results to outweigh the pre-fitting risk and therefore use it in our evaluation.

C. Named-Entity Recognition

For the task of named-entity recognition, we created our own ground-truth dataset to achieve a sufficient dataset size and diversity while avoiding the risk of pre-fitted models. We did so following an approach that we call "search-based indirection of reference results": We start with a substantial set of entities considered to be relevant for NER, feed each of these entities into a search engine, and take the results as input data for the NER services to be evaluated. The original entities, in turn, represent our reference results which a NER service should detect correctly within the input data.

More concretely, we first extract the titles of trending (by descending edit date) Wikipedia articles for the three categories covered by all NER services to be evaluated: person, location, and organization. We assume that each respective Wikipedia page describes one entity. We then search for online news articles that reference this title string in their text using Google News. Finally, we take the first 100 resulting news articles, split their text body into single sentences, and keep those sentences that actually contain the title string (and, thus, the entity to be recognized as reference result).

This leads to a dataset with one entry per named-entity (e.g., *Hugh Jackman*) and for each entry, several sentences containing it (e.g., "Big-screen movie stars Chris Hemsworth and *Hugh Jackman* are headlining...", etc.). Basically, this approach allows to easily auto-generate ground-truth datasets of arbitrary size through varying the length of the editing window (and, thus, the number of initial Wikipedia articles / named entities to be included) as well as the number of news results to be considered for each entity.

Measurand – Accuracy per Entity and per Entity Type: As the primary measurand for evaluating the quality of an NER service, we employ the detection and classification accuracy per entity to be recognized, whereas an accurate classification of an entity obviously presupposes its correct detection. Basically, a request to a NER service can lead to four possible results: (1) the reference entity is detected and correctly classified, (2) the reference entity is detected but not classified correctly, (3) the reference entity is not among the detected ones, and (4) no entities are detected at all. Only in the case of (1), we consider a sentence as correctly processed. Using sets of 5 unique sentences that contain the same reference entity at least once, we define the accuracy *per entity* (e.g., for "Alan Turing") as the proportion of correctly processed sentences out of the 5 ones tested for this entity. If the service's accuracy for a particular entity is 0, we conclude that this entity is unknown to the service. Based on these entity-specific accuracies, we calculate aggregate accuracies – proportions of correctly processed sentences – per entity type (person, location, organization) as well as an overall accuracy for each of the covered services.

Additional Indicators: To support further analysis, we additionally measure the proportion of partially correctly processed sentences in a similar way, assuming a sentence to be partially correctly processed if the detected entity string (e.g., "Elvis") is a substring of the annotated entity string (e.g., "Elvis Aaron Presley") and the detected entity type is correct (e.g., person).

Validity and Limitations: Using the approach described above, we first used the Wikidata query service to identify all Wikipedia entries edited during a time window of 90 days, written in English and marked as describing a person, location, or organization. For each of the resulting reference entries, we retrieved and processed the top 100 Google news entries as described above, resulting in an overall of 139,915 sentences for 3,005 persons, 343,766 sentences for 3,216 locations, and 127,322 sentences for 2,843 organizations. We consider this sufficient in matters of size and variety (R4).[4]

The approach of search-based indirection also provides realistic (R2) and relevant input data (news articles) and associated reference results (persons, locations, and organizations sufficiently prominent for being present and recently edited on Wikipedia – R1) for NER and ensures that reference results are correct outputs for the respective input (they necessarily appear in the test data – R3). Finally, our approach of search-based indirection intentionally introduces a factor of non-determinism and further heightens non-staticness through the editing window. It thereby renders the risk of providers' models being pre-fitted to the exact dataset employed in our experiments sufficiently low (R5).

We see only two limitations applying to our self-generated dataset: First, there is a certain residual risk of formatting errors present in automatically retrieved and cleansed sentences. After manual sample inspection, however, we see this risk as being marginal. Second, our dataset can not be used for evaluating higher-order functions of NER such as detecting references across multiple different terms within one sentence, given the fact that we always have exactly one reference result per sentence. This is, however, not relevant for our experiments as we only focus on the foundational NER functionality of accurately detecting and classifying one particular entity that is known to be present within a sentence. Both limitations do thus not affect the validity of our results.

[4] As mentioned above, both could easily be increased through bigger editing windows and a higher number of news items per reference entity.

D. Text Classification

To evaluate text classification, we use another implementation of the above-mentioned approach of "search-based indirection" to create our own ground-truth dataset. Here, the two cloud providers that offer text classification (Google and IBM) have different numbers of default content categories (Google has 621, IBM has 1037) and different taxonomy depths (Google has 3 levels, IBM has 5). Moreover, the categories they have in common are mostly named differently (e.g., *Autos & Vehicles* vs. *automotive and vehicles*).

This led us to the following approach of gathering ground-truth datasets for appropriately assessing text classification services: First, we scraped all level-1+ content categories (sub-categories, sub-subcategories, etc.) from Google's and IBM's website and generated Twitter Hashtags out of their names. For example, *dermatology* would become *#dermatology*, or *freshwater fishing* would become *#freshwaterfishing*.

Next, we automatically queried Twitter.com for tweets – covering a timespan of 4 weeks – that (i) contain one of these hashtags, (ii) are written in English, and (iii) contain an external URL that refers to either a news website or an online blog. For every result of each query matching these criteria, we then checked whether it actually contained the searched hashtag and skipped it if not. Additionally, we checked if the result contained at least one additional hashtag matching an existing level-1+ category and skipped the respective result in case it does to avoid ambiguities regarding the accuracy of classification. For the remaining results, we then downloaded and parsed the referred website's HTML document, leaving us with an English text (the input data to be tested) that was tagged with exactly one existing level-1+ category (the reference result) by an independent Twitter user. Level-0 category hashtags were consciously left out as categories such as *#society* or *#news* are too general to be considered valid reference results and could lead to misleading texts.

For each item in the resulting, provider-specific sets of ground-truth data, we then tested whether the respective provider's text classification service accurately determines (i) the level-0 category (e.g., *Sports*) for a news article tagged with a level-1+ category (e.g., *Combat Sports* or *Boxing*), and, taking it one step further, (ii) whether it correctly determines the level-1+ category used to generate the search hashtag.

Measurand – Accuracy on Classification level-0 and 1+: For text classification, we use two measurands to represent the accuracy provided by a particular service: If the service correctly detects a news article's level-0 category (e.g., "sports" for an article gathered based on the level-1+ category/hashtag "boxing"), we say the news article was correctly level-0 processed. We call the proportion of correctly level-0 processed news articles (out of all news articles) the level-0 accuracy. If the service also correctly detects a news article's level-1+ category, we say the news article was correctly level-1+ processed and similarly aggregate the service's level-1+ accuracy.

Validity and Limitations: First of all, news articles are realistic input data for text classification (R2). Like for named entity recognition, the approach of search-based indirection inherently produces bipartite dataset entries (R1), consisting of input data (news articles to be classified) and associated reference results (level-1 categories used for hashtag generation and implicitly associated level-0 categories). In matters of size, we gathered 21,393 news articles based on 529 available level-1+ categories (from 24 level-0 categories) for Google's text classification service and 22,255 news articles resulting from 694 level-1+ categories (belonging to 22 level-0 categories) for IBM. We consider this to be sufficient (R4).

For each service, all existing level-1+ categories are covered and linked to different news articles by independent Twitter users, suggesting an appropriate diversity (R4) and relevance. The employed approach also ensures a high correctness of reference results (level-0 and level-1+ categories) for the respective input data (news articles) as this linkage is implicitly provided by independent Twitter users with an inherent interest in proper attributions of hashtags to news articles (R3). Again, our approach of search-based indirection limits the risk of pre-fitted models through non-static search results (R5).

Limitations possibly emanating from multiple category-matching hashtags being used within one tweet were consciously precluded during the collection of ground-truth data. Such cases could otherwise have led to correct matchings returned by a service wrongly being considered inaccurate. Beyond this, and similarly to the task of named-entity recognition, there also exists a residual risk of formatting errors which we do, however, again consider to be negligible after cursory inspection of our datasets.

V. Experiment Execution

For executing our experiments, we implemented an experiment suite consisting of separate modules for (i) gathering above-mentioned ground-truth data, (ii) running experiments for the different NLP tasks across all covered cloud providers, and (iii) analyzing experiment results. The suite was implemented in python, using well-established libraries such as BeautifulSoup or Twython for gathering data or spaCy, NLTK and Enchant for calculating additional indicators (see above) on our ground-truth data. As experiments are comparably long-running, we implemented an experiment execution pipeline capable of interrupting experiment runs and resuming them at the previously reached state. Gathered datasets, as well as experiment states and results, were stored in separate collections of a MongoDB. For each of the evaluated providers, we employed the respective python SDK to execute our queries. The API versions used are Feb 2020 (Google and Microsoft), Mar 2020 (Amazon), and Jul 2019 (IBM). We refer to our GitHub repository[5] for further details.

All experiments were conducted on a LAN-connected off-the-shelf desktop machine located in Germany. Given the accuracy- (as opposed to performance-) oriented nature of experiments, this is a viable approach. Cloud services were used based on academic grants of the respective providers.

[5] https://github.com/dimitristaufer/Cloud-NLP-Evaluation

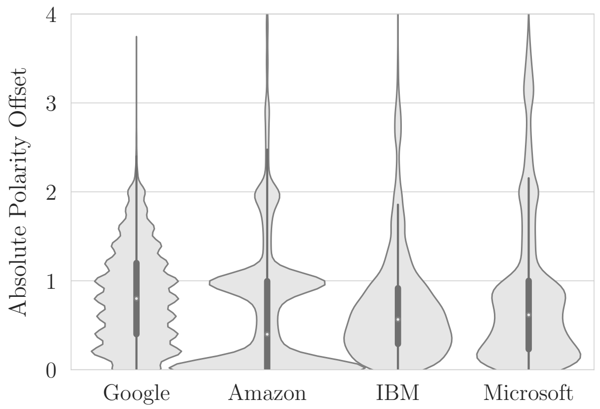

Fig. 2. Violin plot of $APO \in [0, 4]$ distributions (lower is better) delivered by providers for sentiment analysis of the Yelp dataset.

For AWS, this implied using the us-east-1 region, while for Microsoft and IBM we used the regions West Europe and London, respectively. Google makes no further specifications about where NLP services are ultimately executed. All NLP services were used "as is", without any reconfigurations, custom-trained models, etc.

Even though performance factors were clearly not the main scope of our experiments, and even if employed regions differ significantly among providers, we were nonetheless also interested in at least rough estimations about the latencies exhibited by the different services. Due to some malfunctions only discovered after executing our main experiments, we conducted an additional yet smaller line of experiments explicitly dedicated to this. Being well-aware of the limited significance of respective measurements, they do nonetheless provide first indications about latencies to be expected in practice and about possibly existing latency discrepancies between the providers.

VI. Experiment Results & Discussion

Our accuracy-oriented results shall be presented and discussed in the same succession of sentiment analysis, named-entity recognition, and text classification established before. Beyond these, we also briefly look at the additional factors of costs raised by the different providers for executing our exemplary experiments as well as at potentially existing discrepancies in the observed latencies to allow for a more comprehensive view.

A. Sentiment Analysis

As laid out above, the main measurand describing the provided accuracy of sentiment analysis services is the absolute polarity offset (APO) observed between provided results and the reference results included in the Yelp dataset. A smaller APO implies a higher accuracy.

When looking at overall results for the Yelp dataset, we observe only slight differences between the providers: Amazon delivers the lowest median APO (0.39) followed by IBM (0.55), Microsoft (0.61), and Google (0.79). At the same time, Amazon has the biggest interquartile range (IQR) of deviation between the 25th and the 75th percentile (0.99) while IBM (0.63), Microsoft (0.78) and Google (0.79) perform better in this regard. We can thus say that Amazon provides the best median results while IBM achieves the highest result consistency, even though these differences are rather marginal.

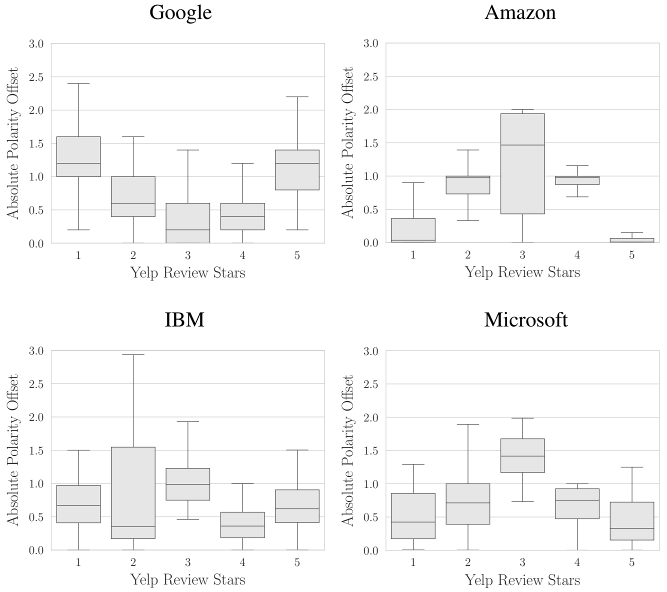

Fig. 3. Boxplots of APOs for sentiment analysis depending on star rating.

When looking at the distributions of APOs (see fig. 2), however, it becomes apparent that Amazon exhibits a strong tendency towards whole numbers. This might indicate some sort of pre-fitting in the sense that inputs are (correctly) recognized as "whole-number reviews". In any case, this distribution calls for further inspection in future work.

Similarly, results are also revealing when looking at the APOs provided for reviews with a particular star rating. Google, for instance, provides very low APOs for reviews with a moderate star rating but is less accurate for extreme reviews with 1 or 5 stars, while Microsoft and Amazon deliver rather inverse results with low median APOs for reviews with a strong bias and comparably high APOs for rather indifferent ones. In addition, Amazon exhibits remarkably low IQRs for positive and negative reviews while being significantly less consistent for moderate 3-star ones. IBM, in turn, provides rather balanced results with mean APOs below 1.0 for each star rating. The only noteworthy effect here are comparably large deviations for 2-star ratings (see figure 3).

Besides dependence on the star rating, we also analyzed the effects of reviews' length, usefulness rating, and percentage of adjectives on the APO. Interestingly, Amazon's median APO is very low for short reviews of 400 characters or less and then continuously increases with higher character counts while other providers delivered rather constant (but, as outlined above, worse) APOs. No provider, however, profited from longer reviews with significantly lower mean APOs.

A comparable effect is also observable for the proportions of adjectives present in a review. We expected that more adjectives allow for a better analysis of the sentiment embodied in a review. This expectation slightly materialized for Google, IBM and Microsoft. Amazon, however, counter-intuitively provides significantly lower median APOs for reviews with a lower proportion of adjectives, albeit with a virtually constant IQR.

The rated usefulness of reviews, in turn, also showed only marginal effects on the result quality for Google, Microsoft, and IBM while Amazon produces higher median APOs (worse results) for higher usefulness ratings as soon as a minimum usefulness is exceeded (again, however, with virtually no observable effect on the IQR). As very short reviews typically do not get high usefulness ratings, this fits well with Amazon's previously mentioned low APOs for shorter reviews and its quite constant IQRs.

Altogether, this suggests a superior accuracy of Amazon's sentiment analysis – especially for shorter reviews and for those carrying a clear opinion – without excessively relying on mere opinion words (which would otherwise have materialized in a dependence on high adjective proportions). When it is more important to correctly identify moderate opinions, Google has clear advantages.

B. Named Entity Recognition

For named entity recognition, we distinguish between exact and partial correctness of results (see IV-C) for persons, locations, and organizations. As no significant differences could be identified between exact and partial correctness, we concentrate on the numbers for exact recognition and categorization here. Overall, Google provides the highest accuracy (0.68) with Amazon (0.64), IBM (0.55), and Microsoft (0.54) following. Google also provides the highest accuracy for persons and locations while organizations were identified and categorized best by Amazon (see table I).

TABLE I
OVERALL AND PER-CATEGORY NER ACCURACIES

	Google	Amazon	IBM	Microsoft
Overall	0.6786	0.6420	0.5481	0.5381
PER	0.8752	0.8167	0.7656	0.7796
LOC	0.5832	0.4879	0.4202	0.4725
ORG	0.6005	0.6508	0.4871	0.3840

Interestingly, all providers recognized persons significantly better than locations or organizations. In addition, all providers except Microsoft performed better for organizations than for locations. With exact recognition rates below 50% for our test dataset, IBM and Microsoft have significant room for improvement in the detection of locations and organizations. With regard to locations, the same is also true for Amazon.

When distinguishing between single- and multi-word entities, all providers produce significantly better results for single- (e.g., "Paris") than for multi-word entities (e.g., "The United States of America"). For instance, Google provided an accuracy of 0.74 for single- and of 0.57 for multi-word entities. Given the marginal differences observed between exact and partial correctness, we conclude that all providers miss a substantial proportion of multi-word entities completely. The decline is, however, more noticeable for Amazon (0.74 vs. 0.47) than for IBM (0.61 vs. 0.42) and Microsoft (0.58 vs. 0.46).

As the above numbers do not illuminate to what extent entities were completely unknown for the different services, we calculated this proportion separately. In doing so, we assume an entity to be unknown to a service when none of the five sentences tested for that entity gets correctly processed. The results, shown in table II, invertedly resemble the observed accuracies: Google has the lowest rates of unknown entities for persons, locations, and overall while Amazon completely misses the fewest organizations and all providers have substantially lower unknown rates for persons than for locations and organizations. To a certain extent, differences in accuracies may thus simply be subject to unequally comprehensive sets of known entities maintained by the providers for the different categories.

TABLE II
OVERALL AND PER-CATEGORY PROPORTIONS OF UNKNOWN ENTITIES

	Google	Amazon	IBM	Microsoft
Overall	0.1486	0.1506	0.2305	0.2872
PER	0.0267	0.0671	0.0827	0.0947
LOC	0.1963	0.2271	0.3095	0.3195
ORG	0.2099	0.1431	0.2836	0.4328

C. Text Classification

Text classification is only offered by two out of the four providers covered herein: Google and IBM. Considering the whole test dataset, results do not differ much among these for both, level-0 and level-1+ accuracies. In both regards, however, IBM (0.58 / 0.35) performs slightly better than Google (0.52 / 0.28).

TABLE III
TC ACCURACIES (OVERALL AND PER GENERALIZED TOPIC)

	L-0 Accuracy		L-1+ Accuracy	
	GCP	**IBM**	**GCP**	**IBM**
Overall	0.60	0.59	0.33	0.36
Culture	0.62	0.56	0.40	0.40
Health & Fitness	0.68	0.81	0.39	0.57
Human Activities	0.69	0.65	0.29	0.39
Science	0.59	0.25	0.30	0.17
Society	0.60	0.69	0.29	0.32
Commercial	0.45	0.49	0.29	0.30

When grouping the providers' level-0 categories into general topics[6] and aggregating observed results within these, however, some significant differences emerge (see table III): For instance, IBM has a significantly higher level-0 and level-1 accuracy for categories falling into the topics of health and fitness while Google on both levels provides significantly more accurate results for the topic of science. This might indicate

[6] Details about the grouping are left out here due to space constraints.

different core areas of intensified training to be focused on by IBM and Google – e.g., it might result from IBM's engagement in Watson Health and Google's commitment to Scholar. A noteworthy deviation is also the rather low level-0 accuracy for the commercial topic. Beyond this, however, differences between Google and IBM as well as between different topics are rather insignificant.

D. Additional Factors

Even though our primary focus is on the qualitative assessment of results provided by the different cloud NLP services, other factors are also relevant in practice. In particular, this includes the costs of the different services (including their pricing models) as well as performance-related questions of latency. Both shall thus be briefly elaborated on to provide a more comprehensive view.

1) Costs: All four providers covered herein bill their NLP services on the basis of units, whereas one unit equals 100 or less unicode characters (with at least 3 units billed per request) for Amazon, 1,000 or less characters for Google and Microsoft, and 10,000 or less characters for IBM. This difference is particularly relevant when a large number of comparably short texts is to be analyzed: A single NLP request for a tweet with a maximum of 280 characters, for instance, would be billed as three units by Amazon and as one unit for Google, Microsoft, and IBM. Put together with higher per-unit prices for those providers with bigger unit sizes and the different per-unit prices providers bill for different services, a sentiment analysis for 100,000 tweets would then, for example, cost \$100 for Google, \$10 for Amazon, \$300 for IBM, and \$200 for Microsoft.

In our experiments, however, we used texts of different length, including very short and very long reviews for sentiment analysis, single sentences for named entity recognition, and whole news articles and blog entries for text classification. These datasets, together with the different pricing models, raised significantly differing costs among the covered providers. These are depicted in table IV.

TABLE IV
COSTS INCURRED FOR EXECUTING EXPERIMENTS

	Google	Amazon	IBM	Microsoft
SEA	\$21.67	\$11.56	\$64.97	\$43.34
NER	\$17.85	\$5.65	\$53.12	\$35.70
TC	\$190.16	N/A	\$72.29	N/A

IBM thus raised the highest costs for our sentiment analysis and named-entity recognition experiments while being clearly advantageous over its only competitor Google for text classification – an obvious consequence of IBM's more coarse unit granularity. Amazon, in turn, exhibited the lowest costs of all providers for sentiment analysis and named entity recognition. Given that these experiments comprise considerable proportions of shorter texts, Amazon's fine-grained pricing scheme provides significant advantages here. For Google and Microsoft, in turn, cost differences for sentiment analysis and named entity recognition emerge from the different per-unit prices alone.[7] Putting these numbers in relation to accuracies may support the choice of a cloud NLP provider for a particular task even further and illuminate which weightings actually have to be made.

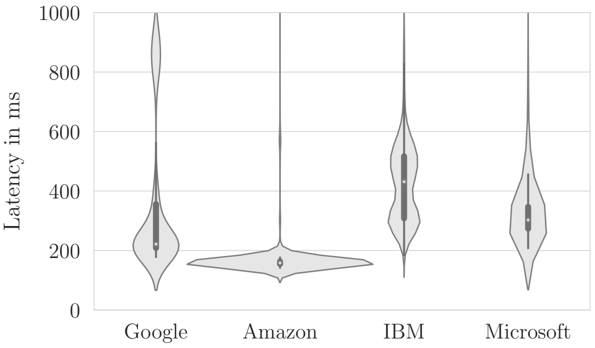

Fig. 4. Violin plot of latency measurements for NER.

2) Latency: Even though our latency-related measurements are of limited significance given the experiment setting (see V), our respective results are worth being reported. Recall that we executed experiments from Germany while the tested services are hosted in different locations, including the US for Amazon, Europe for Microsoft and IBM, and a non-specified location for Google.

Contrary to what could be expected, we observed the lowest median latencies with a remarkably low variance for Amazon. Median latencies here were at 194 ms for sentiment analysis and at 159 ms for named entity recognition, both with IQRs covering less than 32 ms. Microsoft also exhibited comparably stable latencies, albeit with a higher median of 244 ms for sentiment analysis and 303 ms for named entity recognition, respectively. Google and IBM, in turn, deliver less consistent latency values across all tested NLP tasks with Google responding quicker in all cases, and especially for text classification. For an exemplary visualization of latencies observed during named entity recognition, see fig. 4.

While the high latency variance observed for Google may be explained with the undefined hosting region, the remaining differences are still remarkable: IBM exhibits significantly less consistent latencies than Microsoft, even though both are comparable in matters of network distance from the client system. Amazon, in turn, consistently outmatches both in matters of median latency and variance, even though the SUT was hosted in the US. Interestingly, these overall results correspond to similar observations for FaaS platforms [21].

We thus conclude that client-side latencies differ significantly across the four cloud providers' service offers and that network latencies between client and cloud service alone do not explain these observations well. Thus, if low and/or consistent latencies are considered particularly relevant for an application, developers should take different latency charac-

[7] With minimum impact, this also includes varying discounts for high unit numbers.

teristics carefully into account. Besides, our initial screening results motivate additional, more focused experiments.

VII. Conclusion

Cloud NLP services offer various benefits over self-maintained NLP pipelines especially where large amounts of natural language data are to be analyzed ad-hoc or on a recurring basis and where the last bit of functional capabilities is less important than other properties such as low upfront costs or high scalability. However, the accuracy of the results provided by such services has so far been a "known unknown", impairing real-world decisions *whether* to use cloud NLP services instead of self-maintained solutions and, if yes, *which* provider to choose for a particular task to be solved. To a certain extent, the same is also true for the costs and the performance to be expected.

In this paper, we therefore presented a method for the experiment-driven accuracy evaluation of cloud NLP services. We established an experiment design particularly tailored to such services, identified a set of requirements that need to be fulfilled by employed ground-truth datasets, and demonstrated how respective datasets can be gathered. On this basis, we conducted experiments covering NLP services for sentiment analysis, named-entity recognition and text classification offered by the four major cloud providers Amazon, Google, Microsoft, and IBM. In addition, we also examined the costs and the latencies to provide a more comprehensive picture.

Our experiments uncovered significant differences between providers and across examined NLP tasks – in matters of overall accuracies as well as dependencies on the characteristics of input data. Similarly, we also observed significant differences regarding costs and latency behavior. These results already allow for more evidence-based decisions on the above-mentioned questions if and, if yes, which cloud NLP service should be used in a particular case.

Beyond this, our experiment design as well as our method for gathering appropriate ground-truth datasets may serve as blueprints for conducting additional experiments to support so far uncovered decisions. Here, obviously promising strands for future work include the extension to more specialized, NLP-focused cloud providers, more performance-oriented experiments complementing the quality-focused ones conducted herein, similar experiments with even larger or differently generated datasets, and, last but not least, repeatedly re-running our experiments to track future service improvements.

In any case, the work presented herein is, to the best of our knowledge, the first structured, experiment-driven accuracy evaluation of cloud NLP services and provides a valuable basis for real-world decisions as well as for follow-up research.

Acknowledgments

The work behind this paper was partially conducted within the project DaSKITA, supported under grant no. 28V2307A19 by funds of the Federal Ministry of Justice and Consumer Protection (BMJV) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme.

References

[1] B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012, vol. 16.

[2] A. Clark, C. Fox, and S. Lappin, *The handbook of computational linguistics and natural language processing*. John Wiley & Sons, 2013.

[3] M. K. Dalal and M. A. Zaveri, "Automatic text classification: a technical review," *International Journal of Computer Applications*, vol. 28, no. 2, pp. 37–40, 2011.

[4] D. Bermbach, E. Wittern, and S. Tai, *Cloud Service Benchmarking: Measuring Quality of Cloud Services from a Client Perspective*. Cham: Springer International Publishing, 2017.

[5] M. Klems, "Experiment-driven evaluation of cloud-based distributed systems," Ph.D. dissertation, TU Berlin, 2016.

[6] D. Bermbach, J. Kuhlenkamp, A. Dey, A. Ramachandran, A. Fekete, and S. Tai, "Benchfoundry: A benchmarking framework for cloud storage services," in *Proceedings of the International Conference on Service-Oriented Computing*. Cham: Springer, 2017, pp. 314–330.

[7] F. Pallas, D. Bermbach, S. Müller, and S. Tai, "Evidence-based security configurations for cloud datastores," in *Proceedings of the Symposium on Applied Computing*. New York, NY, USA: Association for Computing Machinery, 2017, p. 424–430.

[8] P. Leitner and J. Cito, "Patterns in the chaos: A study of performance variation and predictability in public iaas clouds," *ACM Trans. Internet Technol.*, vol. 16, no. 3, pp. 15:1–15:23, 4 2016.

[9] J. Kuhlenkamp, M. Klems, and O. Röss, "Benchmarking scalability and elasticity of distributed database systems," *VLDB Endowment*, vol. 7, no. 12, pp. 1219–1230, 2014.

[10] J. Kuhlenkamp and M. Klems, "Costradamus: A cost-tracing system for cloud-based software services," in *Proceedings of the International Conference on Service-Oriented Computing*. Cham: Springer, 2017, pp. 657–672.

[11] F. N. Ribeiro, M. Araújo, P. Gonçalves, M. A. Gonçalves, and F. Benevenuto, "Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods," *EPJ Data Science*, vol. 5, no. 1, pp. 1–29, 2016.

[12] D. Zimbra, A. Abbasi, D. Zeng, and H. Chen, "The state-of-the-art in twitter sentiment analysis: A review and benchmark evaluation," *ACM Transactions on Management Information Systems (TMIS)*, vol. 9, no. 2, pp. 1–29, 2018.

[13] S. Atdağ and V. Labatut, "A comparison of named entity recognition tools applied to biographical texts," in *2nd International conference on systems and computer science*. IEEE, 2013, pp. 228–233.

[14] R. Jiang, R. E. Banchs, and H. Li, "Evaluating and combining name entity recognition systems," in *Proceedings of the Sixth Named Entity Workshop*, 2016, pp. 21–27.

[15] Š. Dlugolinskỳ, "Combining named entity recognition methods for concept extraction," *Information Sciences & Technologies: Bulletin of the ACM Slovakia*, vol. 8, no. 2, 2016.

[16] T. Pranckevičius and V. Marcinkevičius, "Comparison of naïve bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing*, vol. 5, no. 2, pp. 221–232, 2017.

[17] Z. H. Moe, T. San, M. M. Khin, and H. M. Tin, "Comparison of naive bayes and support vector machine classifiers on document classification," in *Proceedings of the IEEE Global Conference on Consumer Electronics*, 2018, pp. 466–467.

[18] P. Resnik and J. Lin, *Evaluation of NLP Systems*. John Wiley & Sons, Ltd, 2010, ch. 11, pp. 271–295.

[19] A. Dumitrache, O. Inel, B. Timmermans, C. Ortiz, R.-J. Sips, L. Aroyo, and C. Welty, "Empirical methodology for crowdsourcing ground truth," *arXiv preprint arXiv:1809.08888*, 2018.

[20] S. Vychegzhanin and E. Kotelnikov, "Comparison of named entity recognition tools applied to news articles," in *2019 Ivannikov Ispras Open Conference (ISPRAS)*. IEEE, 2019, pp. 72–77.

[21] J. Kuhlenkamp, S. Werner, M. C. Borges, D. Ernst, and D. Wenzel, "Benchmarking elasticity of faas platforms as a foundation for objective-driven design of serverless applications," in *Proceedings of the Symposium on Applied Computing*. New York, NY, USA: Association for Computing Machinery, 2020, p. 1576–1585.