POLITECHNIKA WARSZAWSKA

Wydział Matematyki i Nauk Informacyjnych

Rozprawa doktorska

Mgr inż. Tomasz Stanisławek

Ekstrakcja informacji z dokumentów o bogatej strukturze graficznej

Promotor
dr hab. inż. Przemysław Biecek

Warszawa 2021

Streszczenie

Bardzo szybki rozwój dziedziny przetwarzania języka naturalnego (ang. *Natural Language Processing*), a w szczególności pojawienie się nowych modeli języka (BERT, RoBERTa, T5, GPT-3) spowodował gwałtowny wzrost skuteczności w rozwiązywaniu standardowych problemów. Wpłynęło to również znacząco na jakość wyników w tematyce ekstrakcji informacji ze zwykłego tekstu. Przykładowo, dla zadania wykrywania jednostek nazewniczych (ang. *Named Entity Recognition, NER*) w samym tylko 2018 roku udało się osiągnąć przyrost o 1.88 pp miary F_1 dla zbioru CoNLL 2003 (wcześniej na taki przyrost trzeba było czekać 11 lat). Te sukcesy spopularyzowały użycie technik ekstrakcji informacji w celu automatyzacji procesów biznesowych, gdzie większość dokumentów posiada bogatą strukturę graficzną.

Celem niniejszej rozprawy doktorskiej jest zbadanie możliwości istniejących metod wykorzystywanych do ekstrakcji informacji z dokumentów o bogatej strukturze graficznej, konceptualizacja problemów, jakie występują w tej dziedzinie, oraz zaproponowanie własnego mechanizmu, który poprawia jakość dotychczasowych rozwiązań.

Wszystkie postawione cele pracy zostały zrealizowane, czego efektem końcowym było utworzenie nowego modelu LAMBERT, który dzięki wstrzyknięciu informacji o pozycji tokenów na stronie osiąga znacząco lepsze wyniki na trzech zbiorach domenowych: Kleister NDA, Charity oraz SROIE.

Abstract

The rapid development of the domain of Natural Language Processing (NLP), and particularly the introduction of new language models (BERT, RoBERTa, T5, GPT-3) resulted in large improvements in solving standard problems. This also substantially influenced the quality of information extraction from plain text. For instance, in case of Named Entity Recognition (NER), only in 2018 an increase of F_1 score on the CoNLL 2003 dataset by 1.88 was achieved. Before that, reaching a similar increase took 11 years! These successes enabled the adoption of information extraction techniques in the field of business process automation, where the majority of documents are not simply plain text, but are endowed with rich layout structure.

The goal of this thesis is the investigation of capabilities of existing methods used for information extraction from visually rich documents, conceptualization of the problems occurring in this field, and finally proposing a new mechanism which improves the quality of hitherto used solutions.

All the presented aims of the thesis have been reached, giving rise to the LAMBERT layout-aware language model. Thanks to injecting the information about token positions on the page, it achieves significant improvements on three datasets from the domain of end-to-end information extraction from visually rivh documents: Kleister-NDA, Kliester-Charity and SROIE.

Podziękowania

> Największą satysfakcję odczuwamy właśnie wtedy, gdy dajemy innym coś z siebie, gdy za cel stawiamy sobie poprawienie warunków życia innych ludzi, gdy przyłączamy się do jakiejś większej sprawy i staramy się wywrzeć pozytywny wpływ na otaczający świat. – Nick Vujicic

Chciałbym bardzo serdecznie podziękować mojemu promotorowi Przemysławowi Bieckowi za wsparcie naukowe oraz motywację, której mi dostarczał w czasie całej naszej współpracy, szczególnie podczas tworzenia niniejszej rozprawy doktorskiej. Wyrazy podziękowania chciałbym również złożyć moim opiekunom naukowym z ramienia firmy: Łukaszowi Garncarkowi, Annie Wróblewskiej oraz Filipowi Gralińskiemu (bez was niniejsza praca również by nie powstała).

Dziękuję również wszystkim współautorom publikacji, z którymi miałem możliwość pracować na różnym etapie mojego rozwoju naukowego. Od wszystkich nich nauczyłem się bardzo dużo i żałuję, że sam często nie mogłem dać od siebie więcej, choćbym chciał. Pragnę również podziękować wszystkim osobom, z którymi miałem okazję rozmawiać o sprawach naukowych, za ich cenne rady i pomysły, którymi mnie obdarzyli (na szczególne wyróżnienie zasługują osoby z laboratorium MI2).

Chciałbym również podziękować wszystkim współpracowaniom z firmy Applica.ai, z którymi miałem okazję pracować (jesteście wspaniali). Dziękują również współtwórcom firmy Adamowi Dancewiczowi i Piotrowi Surmie za możliwość realizacji doktoratu wdrożeniowego w pracy oraz chęć tworzenia firmy, gdzie rozwój pracownika odgrywa ważną rolę.

Na koniec chciałbym podziękować swojej najukochańszej żonie Oldze za cierpliwość, pomoc, troskę oraz mądre rady, którymi mnie obdarza na co dzień. Moim dzieciom: Agacie i Arturowi - praca ta powstawała wspólnym wysiłkiem. Moim rodzicom, rodzeństwu, przyjaciołom za wspólnie spędzone chwile i wyrozumiałość, kiedy nie było na nie czasu.

Spis treści

1 Wprowadzenie

1.1 Motywacja

Ekstrakcja informacji (ang. *Information Extraction, IE*) jest bardzo ważnym aspektem zagadnienia robotyzacji procesów biznesowych (ang. *Robotic Process Automation, RPA*) związanych z przetwarzaniem dokumentów. Przewiduje się, że w 2022 roku cała branża związana z automatyzacją będzie warta ponad 600 miliardów dolarów (w 2020 roku warta była około 480 miliardów dolarów) [39]. Znaczący postęp technologiczny w dziedzinie przetwarzania języka naturalnego (ang. *Natural Language Processing, NLP*) sprawia, że automatyczne przetwarzanie dokumentów jest możliwe i staje się bardzo ważną częścią wspomnianej branży. Istnieje jednak następujący problem: większość dokumentów biznesowych zawiera w sobie nie tylko zwykły tekst, ale również różnego rodzaju struktury (przykładowo: tabele, listy, tekst pogrubiony czy formularze), które uniemożliwiają poprawne przetwarzanie aktualnie istniejącymi metodami (przetwarzającymi tekst w postaci sekwencji tokenów). Na dodatek da się zauważyć, że mechanizmy pojawiające się od początku 2019 roku, uwzględniające nie tylko tekst, ale w jakiś sposób strukturę dokumentów, nie działają lepiej od ostatnio wprowadzonych modeli języka [18, 29, 14, 8, 24].

Nieodzownie zatem pojawia się potrzeba zrozumienia problemów wynikających z ekstrakcji dokumentów o bogatej strukturze graficznej (ang. *Visually Rich Documents, VRDs*) oraz zaproponowania rozwiązania uwzględniającego aspekty struktury dokumentu. Ma to szczególne znaczenie dla rozwoju technologicznego wielu firm, w których obieg informacji odbywa się poprzez różnego rodzaju dokumenty przekazywane w postaci plików PDF, Word itp.

1.2 Opis problemu

1.2.1 Ekstrakcja informacji

Wyobraźmy sobie sytuację, w której ktoś pracujący jako historyk ma za zadanie uporządkować dokumenty o charakterze biograficznym. Przykładowo, mając do dyspozycji tekst[1]:

> Urodził się 29 września 1881 w domu nr 13 przy ulicy Jagiellońskiej we Lwowie, w żydowskiej rodzinie Artura Edlera i Adeli (z domu Landau) von Misesów. Ojciec Ludwiga był absolwentem Politechniki w Zurychu. Potem pracował w austriackim

[1] Źródło: `https://pl.wikipedia.org/wiki/Ludwig_von_Mises`

> Ministerstwie Kolei jako inżynier. Ludwig był najstarszym z trójki chłopców. Jeden z braci umarł w dzieciństwie, natomiast Richard został znanym matematykiem. W latach 1892–1900 Ludwig von Mises uczęszczał do prywatnej szkoły podstawowej we Lwowie.

chce wydobyć informację o imieniu, nazwisku i dacie narodzin opisywanej osoby (w tym wypadku odpowiednio: `Ludwig`, `von Mises`, `1881-09-29`). Taki proces nazywamy ekstrakcją informacji, czyli zamianą danych nieustrukturyzowanych (ang. *unstructured data*) w postaci tekstu do postaci danych posiadających strukturę (ang. *structured data*) [15].

1.2.2 Ekstrakcja informacji z dokumentów o bogatej strukturze graficznej

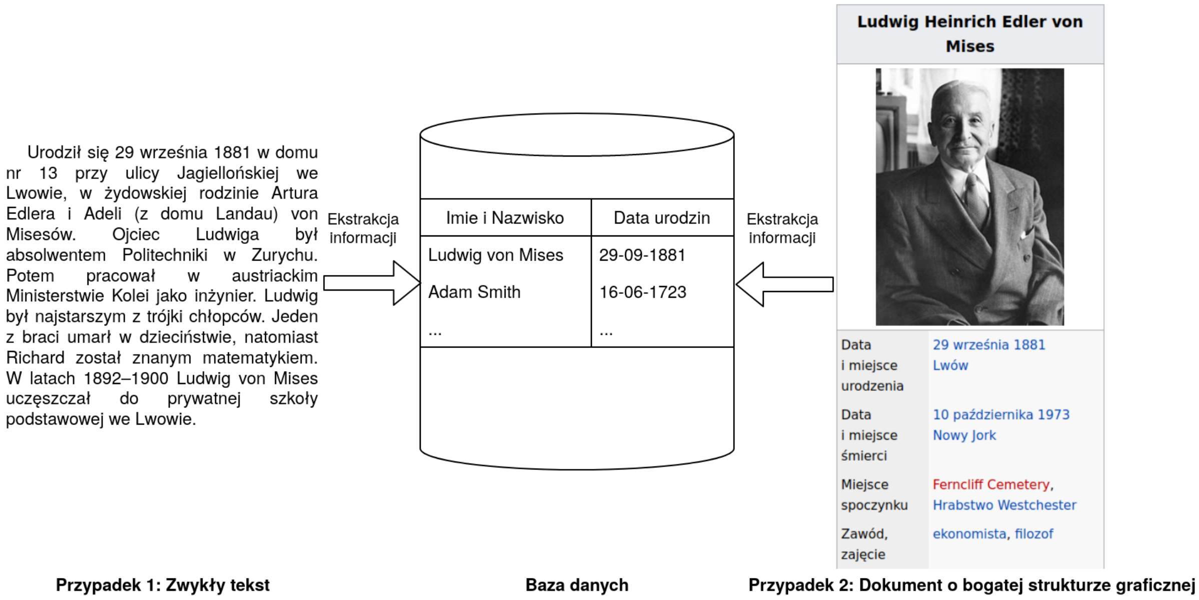
Na Rysunku 1[2] zaprezentowano dwa przypadki ekstrakcji informacji: 1) ze zwykłego tekstu, 2) z dokumentu o bogatej strukturze graficznej. Intuicja podpowiada nam, że do rozumienia tych dwóch zaprezentowanych przykładów potrzebujemy innych umiejętności odczytywania informacji. Mianowicie, w przypadku 1) musimy rozumieć język (strukturę zdania, znaczenie, itp.), podczas gdy w przypadku 2) żeby w pełni zrozumieć treść przekazu, potrzebna nam jest dodatkowo informacja o strukturze tego dokumentu (m.in. układu i relacji względem siebie poszczególnych słów w przestrzeni dwuwymiarowej).

Opisany powyżej przykład 2) jest jedynie prostym ukazaniem różnic pomiędzy wydobywaniem informacji ze zwykłego tekstu a wydobywaniem ich z dokumentu o bogatej strukturze graficznej. Należy zwrócić uwagę, że istnieje bardzo dużo przykładów układów stron, za pomocą których człowiek koduje informacje – szczególnie w domenie biznesowej (Rysunek 2) [54]. W okresie rozpoczęcia badań przedstawianych w niniejszej pracy (2018 rok) nie było jeszcze w świecie nauki wystarczającej świadomości wyzwań, z jakimi należy sobie poradzić, przetwarzając dokumenty o bogatej strukturze graficznej, co również stanowiło problem do rozwiązania.

1.2.3 Zagadnienie powiązane – wstępne przetwarzanie dokumentów

Bardzo duża część dokumentów dostępnych na komputerach ma postać plików PDF (ang. *Portable Document Format*). Z jednej strony wiąże się to z zaletami w postaci uniwersalnego sposobu odczytu danych. Z drugiej strony wadą tego formatu jest to, że nie posiada on żadnych standardów dotyczących sposobu zapisu informacji tekstowych, jego struktury oraz grafiki, która

[2] Źródło: `https://pl.wikipedia.org/wiki/Ludwig_von_Mises`



Urodził się 29 września 1881 w domu nr 13 przy ulicy Jagiellońskiej we Lwowie, w żydowskiej rodzinie Artura Edlera i Adeli (z domu Landau) von Misesów. Ojciec Ludwiga był absolwentem Politechniki w Zurychu. Potem pracował w austriackim Ministerstwie Kolei jako inżynier. Ludwig był najstarszym z trójki chłopców. Jeden z braci umarł w dzieciństwie, natomiast Richard został znanym matematykiem. W latach 1892–1900 Ludwig von Mises uczęszczał do prywatnej szkoły podstawowej we Lwowie.

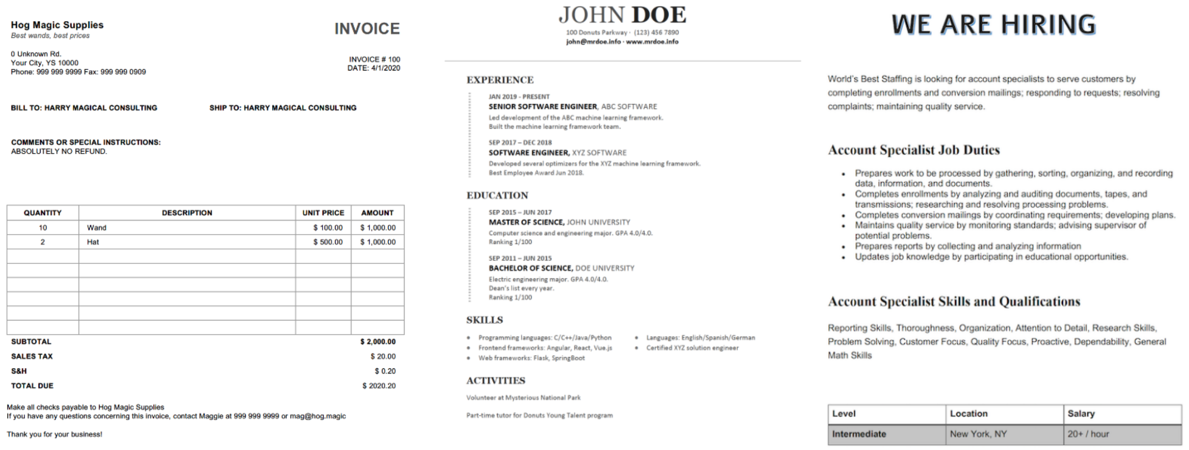
Imie i Nazwisko	Data urodzin
Ludwig von Mises	29-09-1881
Adam Smith	16-06-1723
...	...

Ludwig Heinrich Edler von Mises

Data i miejsce urodzenia	29 września 1881 Lwów
Data i miejsce śmierci	10 października 1973 Nowy Jork
Miejsce spoczynku	Ferncliff Cemetery, Hrabstwo Westchester
Zawód, zajęcie	ekonomista, filozof

Ekstrakcja informacji
Ekstrakcja informacji
Przypadek 1: Zwykły tekst
Baza danych
Przypadek 2: Dokument o bogatej strukturze graficznej

Rysunek 1: Ekstrakcja informacji ze zwykłego tekstu (przypadek 1) i z dokumentu o bogatej strukturze graficznej (przypadek 2).



Hog Magic Supplies
Best wands, best prices

0 Unknown Rd.
Your City, YS 10000
Phone: 999 999 9999 Fax: 999 999 0909

INVOICE

INVOICE # 100
DATE: 4/1/2020

BILL TO: HARRY MAGICAL CONSULTING **SHIP TO: HARRY MAGICAL CONSULTING**

COMMENTS OR SPECIAL INSTRUCTIONS:
ABSOLUTELY NO REFUND.

QUANTITY	DESCRIPTION	UNIT PRICE	AMOUNT
10	Wand	$ 100.00	$ 1,000.00
2	Hat	$ 500.00	$ 1,000.00

SUBTOTAL	**$ 2,000.00**
SALES TAX	$ 20.00
S&H	$ 0.20
TOTAL DUE	$ 2020.20

Make all checks payable to Hog Magic Supplies
If you have any questions concerning this invoice, contact Maggie at 999 999 9999 or mag@hog.magic

Thank you for your business!

JOHN **DOE**
100 Donuts Parkway · (123) 456 7890
john@mrdoe.info · www.mrdoe.info

EXPERIENCE

JAN 2019 - PRESENT
SENIOR SOFTWARE ENGINEER, ABC SOFTWARE
Led development of the ABC machine learning framework.
Built the machine learning framework team.

SEP 2017 – DEC 2018
SOFTWARE ENGINEER, XYZ SOFTWARE
Developed several optimizers for the XYZ machine learning framework.
Best Employee Award Jun 2018.

EDUCATION

SEP 2015 – JUN 2017
MASTER OF SCIENCE, JOHN UNIVERSITY
Computer science and engineering major. GPA 4.0/4.0.
Ranking 1/100

SEP 2011 – JUN 2015
BACHELOR OF SCIENCE, DOE UNIVERSITY
Electric engineering major. GPA 4.0/4.0.
Dean's list every year.
Ranking 1/100

SKILLS

- Programming languages: C/C++/Java/Python
- Frontend frameworks: Angular, React, Vue.js
- Web frameworks: Flask, SpringBoot
- Languages: English/Spanish/German
- Certified XYZ solution engineer

ACTIVITIES

Volunteer at Mysterious National Park

Part-time tutor for Donuts Young Talent program

WE ARE HIRING

World's Best Staffing is looking for account specialists to serve customers by completing enrollments and conversion mailings; responding to requests; resolving complaints; maintaining quality service.

Account Specialist Job Duties

- Prepares work to be processed by gathering, sorting, organizing, and recording data, information, and documents.
- Completes enrollments by analyzing and auditing documents, tapes, and transmissions; researching and resolving processing problems.
- Completes conversion mailings by coordinating requirements; developing plans.
- Maintains quality service by monitoring standards; advising supervisor of potential problems.
- Prepares reports by collecting and analyzing information
- Updates job knowledge by participating in educational opportunities.

Account Specialist Skills and Qualifications

Reporting Skills, Thoroughness, Organization, Attention to Detail, Research Skills, Problem Solving, Customer Focus, Quality Focus, Proactive, Dependability, General Math Skills

Level	Location	Salary
Intermediate	New York, NY	20+ / hour

Rysunek 2: Przykłady dokumentów biznesowych z bogatą strukturą graficzną (faktura, CV, ogłoszenie o pracę).

może być umieszczona na stronie. Dodatkowo, w takich dokumentach mogą pojawiać się treści pochodzące ze skanerów reprezentowane jako obrazki. Opisane problemy rozwiązywane są najczęściej przez narzędzia do optycznego rozpoznawania znaków (ang. *Optical Character Recognition, OCR*) [44]. Na ich wejściu podajemy zazwyczaj plik PDF lub obrazek, a na wyjściu otrzymujemy listę słów (niektóre narzędzia zwracają również listę znaków) wraz z odpowiadającymi im współrzędnymi na stronie oraz numerami stron. Do najpopularniejszych ogólnie dostępnych narzędzi należy Tesseract, którego używaliśmy do rozwiązania problemu wstępnego przetwarzania dokumentów [19].

1.3 Cel pracy

Celem przedstawionej rozprawy doktorskiej było zbadanie możliwości istniejących metod wykorzystywanych do ekstrakcji informacji z dokumentów o bogatej strukturze graficznej, konceptualizacja problemów, jakie występują w dziedzinie oraz zaproponowanie własnego mechanizmu, który poprawia jakość dotychczasowych rozwiązań.

1.4 Struktura rozprawy doktorskiej

Niniejsza rozprawa składa się w głównej mierze ze zbioru artykułów naukowych, które zostały dołączone na końcu pracy w formie dodatku. Publikacje te zostały poprzedzone 4 sekcjami, które wprowadzają czytelnika do tematyki rozprawy.

Sekcja 1. przedstawia motywację, opis problemu oraz cel pracy.

Sekcja 2. zawiera prezentację głównych wyników rozprawy i jest podzielona na cztery podsekcje (każda z nich opiera się na jednym artykule). W podsekcji 2.1 weryfikuję aktualne możliwości istniejących metod. W podsekcji 2.2 opisuję sposób tworzenia nowych zbiorów danych do ekstrakcji informacji. Zaprezentowanie nowego porównania do mierzenia postępów w dziedzinie znajduje się w podsekcji 2.3. Wreszcie w podsekcji 2.4 wprowadzam i opisuję nową architekturę modelu z uwzględnieniem struktury dokumentu.

Sekcja 3. obejmuje prezentację mojego dorobku naukowego.

Sekcja 4. to podsumowanie całości pracy i jej wkładu w rozwój dziedziny.

2 Osiągnięte wyniki

2.1 Zbadanie możliwości istniejących metod

Publikacja, na której oparty jest rozdział: *Tomasz Stanislawek, Anna Wróblewska, Alicja Wójcicka, Daniel Ziembicki, and Przemyslaw Biecek. Named Entity Recognition — Is There a Glass Ceiling? In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 624–633, Hong Kong, China, November 2019. Association for Computational Linguistics.*

Wkład autora rozprawy doktorskiej: pomysł oraz konceptualizacja badania; przygotowanie metodologii badania; implementacja oraz ewaluacja poszczególnych metod przedstawionych w publikacji; przeprowadzenie wszystkich eksperymentów; anotacja zbioru danych oraz kontrola spójności i poprawności całego procesu; analiza wyników; pisanie publikacji.

2.1.1 Wprowadzenie

Metody stosowane do wykrywania jednostek nazwanych są często używane do problemu ekstrakcji informacji z dokumentów [29, 55]). Dodatkowo, bardzo szybki rozwój dziedziny przetwarzania języka naturalnego (wprowadzenie modelu Transformera [51] czy pojawienie się modeli języka [33, 8]) w ostatnich latach spowodował znaczący wzrost skuteczności tychże metod. Przykładowo, wyniki ewaluacji na jednym z najpopularniejszych zbiorów danych dla języka angielskiego CoNLL 2003 wzrosły tylko w roku 2018 z poziomu 91.21 do poziomu 93.09 miary F_1 (wcześniej, taki przyrost skuteczności osiągnięto w ciągu 11 lat) [48, 2, 25, 1]. Biorąc pod uwagę powyższe, istotne stało się zrozumienie, jakie typy błędów zostały rozwiązane przez najnowsze modele, jakie są słabe i mocne strony poszczególnych mechanizmów, i w końcu, co stanowi największe wyzwanie dla wszystkich modeli. Największą wartością tej pracy badawczej było to, że dzięki niej udało się wskazać konkretne problemy, z którymi aktualne modele wciąż sobie nie radzą.

2.1.2 Zaproponowana metodologia

Do weryfikacji możliwości istniejących metod przygotowaliśmy metodologię, która składała się z następujących etapów:

Błędy zbioru danych (DE)	Zależności na poziomie dokumentu (DL)
Błędy anotacji (DE-A) - oczywiste błędy anotacji	**Koreferencja na poziomie dokumentu (DL-CR)** - przypadki występowania obiektu w zdaniu, który występuje również w innym zdaniu w tym samym dokumencie
Literówki słowne (DE-WT) - literówki słowne w jednostce nazwanej	**Struktura dokumentu (DL-S)** - struktura dokumentu odgrywa znaczącą rolę, np. obiekty występują w tabelce
Zła segmentacja na poziomie słowa/zdania (DE-BS) - przypadki, gdzie słowo lub zdanie zostało źle podzielone na segmenty	**Kontekst dokumentu (DL-C)** - przypadki, gdzie potrzebny jest cały kontekst dokumentu, aby prawidłowo zaanotować jednostkę nazwaną
Zależności na poziomie zdania (SL)	Ogólne właściwości (G)
Struktura zdania (SL-S) - syntaktyczne właściwości lingwistyczne stanowiące mocną wskazówkę o danym obiekcie	**Dwuznaczność (G-A)** - przypadki, w których jednostki nazwane są użyte w innym znaczeniu, niż zazwyczaj stosowany
Kontekst zdania (SL-C) - przypadki, gdzie kontekst zdania jest wystarczający	**Niespójność anotacji (G-I)** - różna anotacja na poziomie zbioru dla tych samych jednostek nazwanych
	Trudne przypadki (G-HC) - różna możliwość interpretacji

Tablica 1: Taksonomia z kategoriami lingwistycznymi, przedstawiająca najbardziej prawdopodobne przyczyny błędów

.

1. Selekcja zbioru danych, na którym przeprowadzono badanie: wybrany został angielski zbiór CoNLL z 2003 r., wspomniany we wstępie.
2. Wybranie metod do ekstrakcji jednostek nazwanych, które znacząco przyczyniły się do postępów w dziedzinie: Stanford (CRF), CMU (LSTM-CRF), ELMO, BERT-base oraz Flair [10, 22, 33, 8, 32, 1].
3. Odtworzenie wyników dla wybranych metod wraz z zebraniem błędów (przykładów, na których konkretny model zwrócił nieprawidłowy wynik).
4. Przegląd próbki zebranych błędów w celu zdefiniowania taksonomii opisującej z perspektywy lingwistycznej najbardziej prawdopodobne źródło błędów (Tablica 1).
5. Anotacja każdego przykładu do uprzednio zdefiniowanej taksonomii (jeden przykład może być przypisany do kilku kategorii jednocześnie).
6. Analiza wyników.

Model	DE-WT	DE-BS	SL-S	SL-C	DL-CR	DL-S	DL-C	G-A	G-HC	G-I	Ogółem
Stanford	10	38	46	448	372	202	247	219	72	19	703
CMU	6	39	21	378	316	107	175	183	68	20	554
ELMO	9	33	13	250	198	97	144	98	65	21	395
Flair	8	33	16	223	184	100	146	101	59	20	370
BERT	10	40	11	300	263	117	170	94	65	20	472

Tablica 2: Liczba błędów dla wybranego modelu i kategorii lingwistycznej.

2.1.3 Prezentacja i omówienie wyników

Informacja o liczbie błędów popełnianych przez poszczególne modele znajduje się w Tablicy 2. Pierwszym istotnym wnioskiem płynącym z otrzymanych wyników jest to, że nowoczesne mechanizmy bazujące na modelach języka (trenowane w trybie nienadzorowanym na dużych zbiorach danych) takie jak ELMO, BERT oraz Flair mają najmniej błędów i rozwiązują najwięcej problemów w kategoriach SL-C (kontekst na poziomie zdania) oraz G-A (dwuznaczność słów stanowiących jednostkę nazwaną). Wciąż jednak nie potrafią rozwiązywać ogólnych problemów związanych z trudnymi przypadkami (G-HC), niespójnością anotacji (G-I) oraz literówkami językowymi (DE-WT). Wspomniane błędy wraz z oczywistymi pomyłkami anotacji (DE-A) wskazują na jeden z możliwych kierunków rozwoju tej dziedziny, czyli budowy narzędzi do walidacji poprawności zbioru danych. Dodatkowo, tworzenie wolnych od błędów zbiorów testowych będzie poprawiać walidację różnych modeli, szczególnie, jeśli różnice między nimi są niewielkie.

Wszystkie przebadane rozwiązania w tamtym okresie bazowały wyłącznie na kontekście zdania. Natomiast wysoka liczba błędów popełnianych przez modele na poziomie zależności dokumentu (DL-CR, DL-C, DL-S) podpowiada, że żeby osiągnąć jeszcze lepsze wyniki, musimy budować rozwiązania bazujące na całym kontekście dokumentu.

Rozpatrywany zbiór danych zawierał w sobie dokumenty z wiadomości sportowych lub raportów giełdowych, w których struktura dokumentu (dane umieszczone w tabelce lub nagłówku) odgrywa istotną rolę w zrozumieniu treści. Wysoka liczba błędów w kategorii związanej ze strukturą dokumentu (DL-S) informuje nas o kolejnym potencjalnym kierunku rozwoju dziedziny, jakim jest uwzględnienie tychże informacji (np. pozycji słów na stronie). Należy wspomnieć, że w tamtym okresie istniały już próby ekstrakcji informacji z uwzględnieniem pozycji na stronie, ale bazowały one głównie na technikach pochodzących z dziedziny wizji komputerowej, przez co nie dawały tak dobrych wyników jak stosowane wówczas modele języka [18, 23, 13].

2.1.4 Znaczenie przeprowadzonego badania

W poprzednim rozdziale omówione zostały główne wyniki i wnioski przeprowadzonego badania oraz wynikające z nich możliwe kierunki rozwoju: budowa narzędzi wykrywających błędy zbioru danych, uwzględnianie całego kontekstu dokumentu czy branie pod uwagę struktury dokumentu. Z perspektywy rozwiązania problemu ekstrakcji informacji z dokumentów o bogatej strukturze graficznej najbardziej pożądanym kierunkiem było uwzględnienie struktury dokumentu (np. pozycji poszczególnych słów na stronie), w związku z czym dalsza część przeprowadzonych badań skupiła się na tym zagadnieniu.

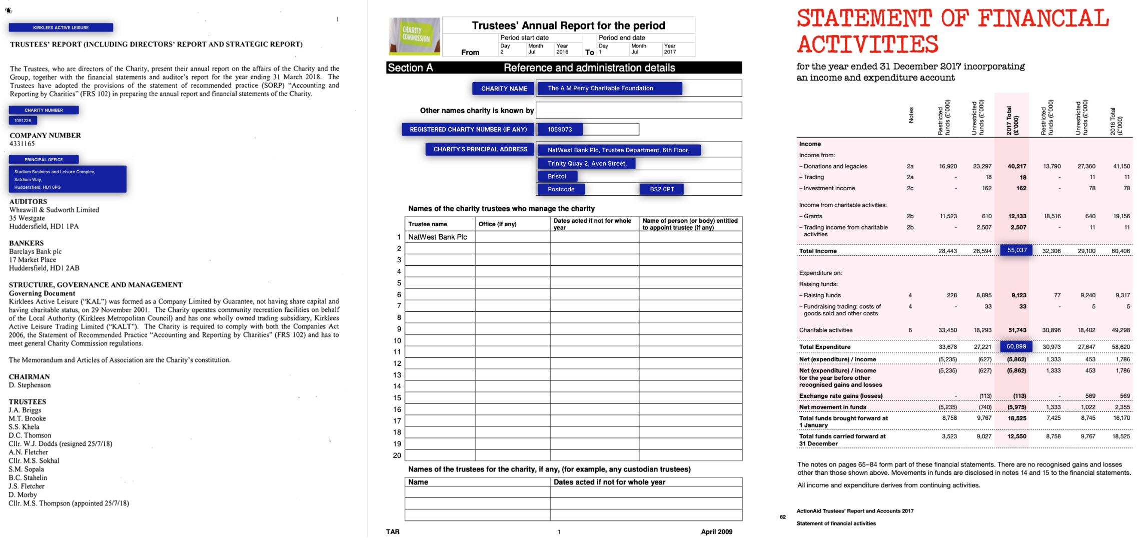
2.2 Kleister – zbiory danych do testowania nowych metod

Publikacja, na której oparty jest rozdział: *Tomasz Stanisławek, Filip Graliński, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. Kleister: Key information extraction datasets involving long documents with complex layouts. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, Document Analysis and Recognition – ICDAR 2021, pages 564–579, Cham, 2021. Springer International Publishing.*

Wkład autora rozprawy doktorskiej: konceptualizacja oraz przygotowanie metodologii pracy; implementacja oraz ewaluacja poszczególnych metod przedstawionych w publikacji; przeprowadzenie wszystkich eksperymentów; analiza wyników; pisanie publikacji; pisanie odpowiedzi do recenzentów.

2.2.1 Wprowadzenie

Opracowanie nowych metod do ekstrakcji informacji dla dokumentów o bogatej strukturze graficznej wymaga dostępności dopasowanych do problemu zbiorów danych, na których można testować ich skuteczność. W połowie 2019 roku istniał tylko jeden publiczny zbiór SROIE (ekstrakcja informacji ze skanów paragonów), który jednak nie uwzględniał wszystkich problemów domenowych (np. przetwarzanie dokumentów wielostronicowych) [13]. Ponadto większość metod, która powstawała w tamtym okresie, była testowana na prywatnych zbiorach danych [29, 18, 23]. Biorąc pod uwagę powyższe, zaistniała potrzeba utworzenia publicznego zbioru danych. Praca tu omawiana wprowadza dwa angielskie zbiory danych: Kleister NDA i Kleister Charity (Rysunek 3). Ponadto szczegółowo przedstawione są dodatkowe wyzwania, przed którymi stoi dziedzina ekstrakcji kluczowych informacji. Wszystkie dokumenty udostępnione są w postaci plików PDF, co umożliwi wykorzystanie wielu modalności przez model (tekst, pozycje tokenów w tekście czy obrazek odpowiadający konkretnej stronie).



KIRKLEES ACTIVE LEISURE

1

TRUSTEES' REPORT (INCLUDING DIRECTORS' REPORT AND STRATEGIC REPORT)

The Trustees, who are directors of the Charity, present their annual report on the affairs of the Charity and the Group, together with the financial statements and auditor's report for the year ending 31 March 2018. The Trustees have adopted the provisions of the statement of recommended practice (SORP) "Accounting and Reporting by Charities" (FRS 102) in preparing the annual report and financial statements of the Charity.

CHARITY NUMBER
1091226

COMPANY NUMBER
4331165

PRINCIPAL OFFICE
Stadium Business and Leisure Complex,
Satdium Way,
Huddersfield, HD1 6PG

AUDITORS
Wheawill & Sudworth Limited
35 Westgate
Huddersfield, HD1 1PA

BANKERS
Barclays Bank plc
17 Market Place
Huddersfield, HD1 2AB

STRUCTURE, GOVERNANCE AND MANAGEMENT
Governing Document
Kirklees Active Leisure ("KAL") was formed as a Company Limited by Guarantee, not having share capital and having charitable status, on 29 November 2001. The Charity operates community recreation facilities on behalf of the Local Authority (Kirklees Metropolitan Council) and has one wholly owned trading subsidiary, Kirklees Active Leisure Trading Limited ("KALT"). The Charity is required to comply with both the Companies Act 2006, the Statement of Recommended Practice "Accounting and Reporting by Charities" (FRS 102) and has to meet general Charity Commission regulations.

The Memorandum and Articles of Association are the Charity's constitution.

CHAIRMAN
D. Stephenson

TRUSTEES
J.A. Briggs
M.T. Brooke
S.S. Khela
D.C. Thomson
Cllr. W.J. Dodds (resigned 25/7/18)
A.N. Fletcher
Cllr. M.S. Sokhal
S.M. Sopala
B.C. Stahelin
J.S. Fletcher
D. Morby
Cllr. M.S. Thompson (appointed 25/7/18)

CHARITY COMMISSION

Trustees' Annual Report for the period

	Period start date				Period end date		
From	Day 2	Month Jul	Year 2016	To	Day 1	Month Jul	Year 2017

Section A Reference and administration details

CHARITY NAME: The A M Perry Charitable Foundation

Other names charity is known by:

REGISTERED CHARITY NUMBER (IF ANY): 1059073

CHARITY'S PRINCIPAL ADDRESS: NatWest Bank Plc, Trustee Department, 6th Floor,
Trinity Quay 2, Avon Street,
Bristol
Postcode: BS2 0PT

Names of the charity trustees who manage the charity

	Trustee name	Office (if any)	Dates acted if not for whole year	Name of person (or body) entitled to appoint trustee (if any)
1	NatWest Bank Plc			
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Names of the trustees for the charity, if any, (for example, any custodian trustees)

Name	Dates acted if not for whole year

TAR 1 April 2009

STATEMENT OF FINANCIAL ACTIVITIES

for the year ended 31 December 2017 incorporating
an income and expenditure account

	Notes	Restricted funds (£'000)	Unrestricted funds (£'000)	2017 Total (£'000)	Restricted funds (£'000)	Unrestricted funds (£'000)	2016 Total (£'000)
Income							
Income from:							
– Donations and legacies	2a	16,920	23,297	**40,217**	13,790	27,360	41,150
– Trading	2a	-	18	**18**	-	11	11
– Investment income	2c	-	162	**162**	-	78	78
Income from charitable activities:							
– Grants	2b	11,523	610	**12,133**	18,516	640	19,156
– Trading income from charitable activities	2b	-	2,507	**2,507**	-	11	11
Total Income		28,443	26,594	55,037	32,306	29,100	60,406
Expenditure on:							
Raising funds:							
– Raising funds	4	228	8,895	**9,123**	77	9,240	9,317
– Fundraising trading: costs of goods sold and other costs	4	-	33	**33**	-	5	5
Charitable activities	6	33,450	18,293	**51,743**	30,896	18,402	49,298
Total Expenditure		33,678	27,221	60,899	30,973	27,647	58,620
Net (expenditure) / income		(5,235)	(627)	**(5,862)**	1,333	453	1,786
Net (expenditure) / income for the year before other recognised gains and losses		(5,235)	(627)	**(5,862)**	1,333	453	1,786
Exchange rate gains (losses)		-	(113)	**(113)**	-	569	569
Net movement in funds		(5,235)	(740)	**(5,975)**	1,333	1,022	2,355
Total funds brought forward at 1 January		8,758	9,767	**18,525**	7,425	8,745	16,170
Total funds carried forward at 31 December		3,523	9,027	**12,550**	8,758	9,767	18,525

The notes on pages 65–84 form part of these financial statements. There are no recognised gains and losses other than those shown above. Movements in funds are disclosed in notes 14 and 15 to the financial statements.

All income and expenditure derives from continuing activities.

62 ActionAid Trustees' Report and Accounts 2017
Statement of financial activities

Rysunek 3: Przykłady stron ze zbiorów Kleister NDA i Charity. Niebieskie prostokąty wskazują obiekty do ekstrakcji.

2.2.2 Podstawowe informacje i statystyki zbiorów danych

Zbiór Kleister NDA składa się z dokumentów utworzonych cyfrowo (nie wymagają kroku OCR-a) zawierających umowy o zachowaniu poufności, które pochodzą z bazy EDGAR[3]. Łącznie udało się przygotować zbiór 540 dokumentów (zbiór trenujący: 254, walidadyjny: 83, testowy: 203) składających się z 3 229 stron. W przypadku drugiego zbioru, Kleister Charity, dokumenty zawierają sprawozdania roczne organizacji charytatywnych i zostały ściągnięte bezpośrednio ze strony `https://register-of-charities.charitycommission.gov.uk/` w postaci plików PDF. Zbiór Charity składa się z 2 788 dokumentów (zbiór trenujący: 1729, walidadyjny: 440, testowy: 609), co przekłada się na 61 643 stron.

Do każdego dokumentu została przypisana lista obiektów, które należy z niego wyciągnąć. Nie jest to więc typowe zadanie rozpoznawania jednostek nazewniczych, ponieważ mamy informację o obiekcie na poziomie dokumentu, a nie na poziomie tekstu (nie wiemy, gdzie dokładnie ta informacja w dokumencie występuje). W Tablicy 3 pokazane są podstawowe statystyki dla

[3] `https://www.sec.gov/edgar.shtml`

obu zbiorów wraz z podaniem przykładów wartości obiektów do ekstrakcji (wszystkie obiekty zostały znormalizowane do standardowej postaci zgodnej z jego typem).

Zbiór	Nazwa obiektu	Typ obiektu	Liczba	Liczba unikalnych	Przykład obiektu
NDA	party	OGRANIZACJA/OSOBA	1,035	912	Ajinomoto Althea Inc.
	jurisdiction	LOKALIZACJA	531	37	New York
	effective_date	DATA	400	370	2005-07-03
	term	CZAS TRWANIA	194	22	P12M
Charity	post_town	ADRES	2,692	501	BURY
	postcode	ADRES	2,717	1,511	BL9 ONP
	street_line	ADRES	2,414	1,353	42-47 MINORIES
	charity_name	ORGANIZACJA	2,778	1,600	Mad Theatre Company
	charity_number	NUMER	2,763	1,514	1143209
	report_date	DATA	2,776	129	2016-09-30
	income	KWOTA	2,741	2,726	109370.00
	spending	KWOTA	2,731	2,712	90174.00

Tablica 3: Wykaz obiektów do ekstrakcji dla zbiorów Kleister NDA i Charity.

2.2.3 Zaproponowany mechanizm do ekstrakcji informacji

Kluczowym elementem zaproponowanej metody jest wykorzystanie takich samych modeli, jakie stosuje się do rozpoznawania jednostek nazewniczych. W zadaniu ekstrakcji kluczowych informacji posiadamy anotacje jedynie na poziomie dokumentu, dlatego też musimy dodać moduły wspierające, które pozwolą nam wykorzystać możliwości tagera sekwencyjnego. Rysunek 4 przedstawia cały schemat zaimplementowanego mechanizmu, na który składają się następujące elementy:

1. Moduł do przetwarzania plików PDF - każdy plik PDF musi być zamieniony do postaci pliku json zawierającego informację o słowach, ich pozycjach na stronie oraz numerze strony, z której pochodzą. Dla całego zbioru przetestowane zostały różne silniki stosowane do optycznego rozpoznawania znaków: Azure CV, Tesseract oraz Textract[4][5][6]. Dodatkowo, dla dokumentów ze zbioru Kleister NDA użyto narzędzia djvu, ponieważ

[4] `https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-recognizing-text` - użyto wersji 3.0

[5] `https://github.com/tesseract-ocr/tesseract` - użyto wrsji 4.1.1

[6] `https://aws.amazon.com/textract/` - użyto wersji z 1 Marca 2020

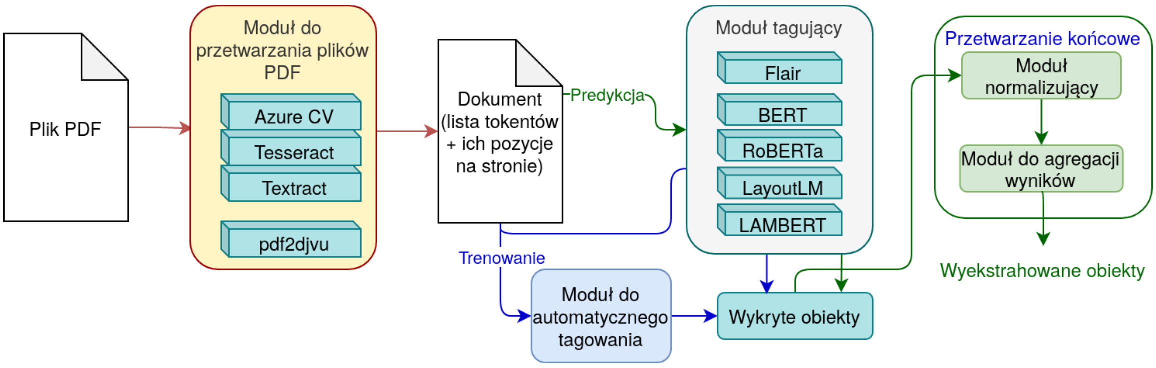
zostały one utworzone cyfrowo [7].

2. Moduł do automatycznego tagowania - potrzebny na etapie trenowania w celu automatycznego oznaczania jednostek nazewniczych na poziomie tekstu, tak aby moduł tagujący mógł się nauczyć. To narzędzie oparte zostało o zestaw wyrażeń regularnych, które wykrywa określoną wartość na różne sposoby w zależności od typu obiektu.

3. Moduł tagujący - bazuje na algorytmach do rozpoznawania jednostek nazewniczych, w którym przetestowane zostały następujące modele: Flair, BERT, RoBERTa, LayoutLM, LAMBERT [1, 8, 24, 57, 11] (Sekcja 2.4).

4. Moduł normalizujący - sprowadza wszystkie obiekty wykryte przez moduł tagujący do postaci standardowej dla każdego typu danych. Przykładowo, wszystkie obiekty typu *DATA* zostaną znormalizowane do postaci zgodnej ze standardem ISO 8601 (fragmenty tekstu *November 29, 2019* oraz *11/29/19* zostaną znormalizowane do postaci *2019-11-29*).

5. Moduł agregujący - mechanizm do wyboru ostatecznej odpowiedzi zwracanej przez system bazujący na danych z poprzedniego modułu. Jest on potrzebny z racji tego, że moduł tagujący może zwracać więcej niż jedną odpowiedź.



Rysunek 4: Schemat ekstrakcji kluczowych informacji składający się z kilku modułów. Wyróżniamy dwa etapy: treningu (przepływ informacji oznaczony kolorem niebieskim) oraz predykcji (przepływ informacji oznaczony kolorem zielonym).

2.2.4 Prezentacja i omówienie wyników

Wyniki ewaluacji poszczególnych modeli zaprezentowano w Tablicy 4. Różnice w jakości działania pomiędzy najlepszym modelem a człowiekiem pokazują trudność tego zadania. Podczas

[7] `http://jwilk.net/software/pdf2djvu`, `https://github.com/jwilk/ocrodjvu`

Zbiór	Nazwa narzędzia	**Flair**	**BERT**	**RoBERTa**	**LayoutLM**	**LAMBERT**	Człowiek
NDA	Azure CV	$78.03_{\pm 0.12}$	$77.67_{\pm 0.18}$	$79.33_{\pm 0.68}$	$77.43_{\pm 0.29}$	$80.57_{\pm 0.25}$	97.86%
	pdf2djvu	$77.83_{\pm 0.26}$	$78.20_{\pm 0.17}$	$81.00_{\pm 0.05}$	$78.47_{\pm 0.76}$	$\mathbf{81.77_{\pm 0.09}}$	
	Tesseract	$76.57_{\pm 0.49}$	$76.60_{\pm 0.30}$	$77.81_{\pm 0.97}$	$77.70_{\pm 0.48}$	$81.03_{\pm 0.23}$	
	Textract	$77.37_{\pm 0.08}$	$74.83_{\pm 0.45}$	$79.49_{\pm 0.32}$	$77.40_{\pm 0.40}$	$77.37_{\pm 0.08}$	
Charity (*)	Azure CV	$81.17_{\pm 0.12}$	$78.33_{\pm 0.08}$	$81.50_{\pm 0.23}$	$81.53_{\pm 0.23}$	$\mathbf{83.57_{\pm 0.29}}$	97.45%
	Tesseract	$72.87_{\pm 0.81}$	$71.37_{\pm 1.25}$	$76.23_{\pm 0.15}$	$77.53_{\pm 0.20}$	$81.50_{\pm 0.07}$	
	Textract	$78.03_{\pm 0.12}$	$73.30_{\pm 0.43}$	$80.08_{\pm 0.15}$	$80.23_{\pm 0.41}$	$82.97_{\pm 0.21}$	

Tablica 4: Szczegółowe wyniki modeli oraz różnych narzędzi do przetwarzania plików PDF dla zbioru testowego uśrednione na podstawie 3-krotnego uruchomienia eksperymentu (zastosowano miarę F_1 z uwzględnieniem standardowego odchylenia). Poziom skompilowania zdania dla człowieka mierzony jest jako procent zgodności anotacji dwóch osób na próbce 100 dokumentów. (*) pdf2djvu nie działa na dokumentach skanowanych.

gdy inne zadania z tej dziedziny są już praktycznie rozwiązane (na zbiorze SROIE najlepszy model osiąga wynik na poziomie 98.36 miary F_1[8]) [13] wprowadzony zbiór przynosi dodatkowe wyzwania, do których należy zaliczyć:

- **Anotacje na poziomie dokumentu.** W związku z brakiem informacji o pozycji obiektów w tekście należy budować dodatkowe moduły wspomagające prace tagerów sekwencyjnych lub przygotować rozwiązania działające w oparciu o model typu seq2seq [46, 37].

- **Bogata struktura graficzna.** Modele działające wyłącznie w oparciu o tekst (Flair, BERT oraz RoBERTa) działają wyraźnie słabiej od modeli uwzględniających również informacje o strukturze dokumentu (LayaoutLM bazujący na modelu BERT oraz LAMBERT bazujący na modelu RoBERTa). Szczególnie widoczne jest to dla obiektów, które znajdują się w tabelkach lub formularzach.

- **Optyczne rozpoznawanie znaków.** Wybór narzędzia do przetwarzania plików PDF był bardzo istotny w kontekście jakości ekstrakcji. Okazuje się, że w niektórych przypadkach był on bardziej istotny niż sam wybór modelu. Najlepszym narzędziem z przebadanych okazał się Azure CV.

- **Długie dokumenty.** Na bardzo długich dokumentach zaobserwowano znaczące pogorszenie się skuteczności modelu [43].

[8] Stan na dzień 25 września 2021 roku.

2.3 DUE – benchmark do mierzenia postępów w dziedzinie rozumienia dokumentów

Publikacja, na której oparty jest rozdział: *Łukasz Borchmann*, Michał Pietruszka*, Tomasz Stanislawek*, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. DUE: End-to-end document understanding benchmark. 2021. (*) Równy wkład w wykonaną pracę.*

Komentarz: Publikacja jest w trakcie recenzji na konferencję NeurIPS 2021 (recenzje można zobaczyć na stronie `https://openreview.net/forum?id=rNs2FvJGDK`).

Wkład autora rozprawy doktorskiej: konceptualizacja i metodologia badania (udział w stałych spotkaniach zespołu); przegląd literatury i przygotowania listy potencjalnych zbiorów, które można wykorzystać; zaprojektowanie schematu danych do przechowywania danych w ujednoliconym formacie; przygotowanie danych dla zbiorów Kleister Charity, DeepForm, Tabfact; poprawa zbiorów danych PWC oraz DeepForm; metodologia utworzenia zbiorów diagnostycznych; organizacja i kontrola anotacji; analiza wyników; poprawa pierwszej wersji publikacji.

2.3.1 Wprowadzenie

Zaprezentowane w sekcji 2.2 zbiory Kleister tyczyły się wyłącznie tematyki ekstrakcji kluczowych informacji, która to jest częścią większej dziedziny rozumienia dokumentów (ang. *Document Understanding*). W tym rozdziale omówiony zostanie benchmark opracowany z myślą o wszystkich zadaniach związanych z przetwarzaniem dokumentów o bogatej strukturze graficznej. Tego typu techniki mierzenia postępów wybranego zagadnienia stały się bardzo popularne, nie tylko w dziedzinie NLP, ale również w innych dziedzinach, takich jak rozumienie obrazów [53, 52, 58, 7]. Dzięki tej pracy naukowcy mają możliwość bardziej generycznego rozwiązywania problemów związanych z przetwarzaniem dokumentów o bogatej strukturze graficznej. Do najważniejszych osiągnięć tej pracy należy zaliczyć:

1. Przegląd listy zadań związanych z dziedziną oraz przeformułowanie trzech zadań: PWC, WTQ, TabFact.

	Liczność (tys.)			Wkład do zbiorów							
Nazwa zbioru	Zbiór trenujący	Zbiór walidujący	Zbiór testowy	Obliczenie trudności	Poprawa zbioru	Przeformułowanie	Poprawa podziału	Ujednolicenie formatu	Zbiór diagnostyczny	Domena	Typ zadania
Kleister Charity [43]	1.73	0.44	0.61	−	−	−	−	+	+	Finansowa	Ekstrakcja informacji z długich dokumentów
PWC [17]	0.2	0.06	0.12	+	+	+	+	+	+	Naukowa	Ekstrakcja informacji z publikacji naukowych
DeepForm [47]	0.7	0.1	0.3	+	+	−	+	+	+	Finansowa	Ekstrakcja informacji z dokumentów biznesowych
DocVQA [27]	10.2	1.3	1.3	−	−	−	−	+	+	Biznesowa	Zadawanie pytań na dokumentach biznesowych
InfographicsVQA [26]	4.40	0.50	0.60	−	−	−	−	+	+	Ogólna	Zadawanie pytań na infografikach
TabFact [5]	13.2	1.7	1.7	−	−	+	−	+	+	Ogólna	Problem *Natural Language Inference* na tabelkach
WTQ [31]	1.4	0.3	0.4	+	−	+	+	+	+	Ogólna	Zadawanie pytań na tabelkach

Tablica 5: Zestawienie wybranych zbiorów danych wraz z ich podstawowymi właściwościami i wkładem do zbiorów.

2. Wyczyszczenie zbiorów danych wraz z przygotowaniem jednolitego formatu do ich przechowywania i procesowania.

3. Implementacja podstawowych metod weryfikacji jakości modeli.

4. Identyfikacja wyzwań, przed jakimi stoi dziedzina, m.in. poprzez przygotowanie zbiorów diagnostycznych.

2.3.2 Wybrane zbiory danych

Podczas tworzenia benchmarku przejrzane zostało ponad 30 zbiorów danych, z których wybrano 7 spełniających kryteria najwyższej jakości (tylko anotacja manualna), trudności (różnica między najlepszym rozwiązaniem a poziomem trudności dla człowieka musi być duża) oraz licencji (Tablica 5) [4]. Uwzględnione zbiory pokrywają szerokie spektrum zdań, począwszy od ekstrakcji informacji, a skończywszy na zadawaniu pytań na tabelkach. Zebrane i poprawione zbiory stanowią główny wkład tej pracy, gdyż dzięki nim pozostali naukowcy mogą skupić się wyłącznie na przygotowaniu nowych rozwiązań.

2.3.3 Zbiory diagnostyczne

Poza przygotowaniem zestawu zbiorów danych z domeny rozumienia dokumentów został utworzony zbiór diagnostyczny wspomagający analizę poszczególnych modeli w konkretnych kategoriach problemu. Szczególnie ważne jest to, byśmy rozumieli działanie poszczególnych modeli, tak aby wiedzieć, co jest silną, a co słabą stroną nowego modelu [40]. Zaproponowana taksonomia wygląd następująco:

- Źródło odpowiedzi (rozumiane jako relacja pomiędzy odpowiedzią a treścią dokumentu): ekstraktywne lub abstrakcyjne.
- Format wyjścia: lista odpowiedzi lub pojedyncza odpowiedź.
- Typ wyjścia: organizacja, lokalizacja, osoba, numer, data/czas/czas trwania, odpowiedź tak/nie.
- Źródło wskazówki do udzielenia odpowiedzi: tabela/lista, czysty tekst, element graficzny, struktura dokumentu, pismo odręczne.
- Operacje: zliczanie, arytmetyka, porównanie, normalizacja.
- Liczba odpowiedzi (ile razy szukana wartość wystąpiła w dokumencie): jednokrotnie lub wielokrotnie.

2.3.4 Przedstawienie i omówienie wyników

Do przeprowadzenia eksperymentów wybrano prosty i wydajmy model T5 [37] oraz jego usprawnienia polegające na dodaniu relatywnego kodowania 2D (model T5+2D) [34, 11, 56]. Oba te modele dodatkowo wytrenowano na dokumentach z domeny biznesowej, o bardziej skomplikowanej strukturze dokumentów (modele T5+U i T5+2D+U).

Wszystkie wyniki, razem z odwołaniem się do najlepszego zewnętrznego wyniku oraz poziomu trudności zadania dla człowieka znajdują się w Tablicy 6. Występują znaczące różnice pomiędzy uzyskanymi wynikami a poziomem, z jakim człowiek radzi sobie z danym zdaniem. Dodatkowo nasze modele nieznaczenie odbiegają od jakości najlepszych wyników, jakie udało się osiągnąć. Dzieje się tak, ponieważ były one dostosowane do konkretnego problemu oraz uwzględniały dodatkowo komponent wizyjny, który nie jest tak prosto zaimplementować.

Największy przyrost osiągnięty został dzięki przetrenowaniu modelu na dokumentach o bogatej strukturze graficznej (T5+U oraz T5+2D+U). Podejrzewamy, że oryginalny model T5 nie zawierał w zbiorze trenującym zbyt wiele skomplikowanych struktur (tabelki czy formularze), przez co nie radzi sobie z nimi. Ponadto widoczna jest poprawa dzięki dodaniu informacji o relatywnej pozycji pomiędzy tokenami, co tylko potwierdza ważność cech strukturalnych przy działaniu modelu.

Zbiór danych	Wynik (metryka uzależniona od zadania)						
	T5	T5+2D	T5+U	T5+2D+U	Najlepszy zewnętrzny wynik		Człowiek
DocVQA	72.5	74.1	76.4	81.3	87.1	[34]	98.1
InfographicsVQA	37.8	43.1	37.0	46.1	61.2	[34]	98.0
Kleister Charity	57.9	57.7	75.1	75.9	83.6	[11]	97.5
PWC	24.2	25.2	25.1	27.3	—		51.1
DeepForm	73.4	74.8	82.0	83.2	—		98.5
WTQ	32.5	33.4	38.1	44.0	—		76.7
TabFact	52.2	53.7	67.9	70.6	—		92.1
Razem	49.8	51.6	56.8	64.4	n/a		88.3

Tablica 6: Najlepsze wyniki dla wybranego modelu w relacji do poziomu trudności dla człowieka oraz najlepszego wyniku osiąganego przez dopasowany do zadania model. Znak — jest wstawiany w miejsce, gdzie nie ma żadnego wyniku, ponieważ zbiór został zmodyfikowany.

2.4 LAMBERT – nowy model do ekstrakcji informacji

Publikacja, na której oparty jest rozdział: *Łukasz Garncarek*, Rafał Powalski*, Tomasz Stanisławek*, Bartosz Topolski*, Piotr Halama, Michał Turski, and Filip Graliński. Lambert: Layout-aware language modeling for information extraction. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, Document Analysis and Recognition – ICDAR 2021, pages 532–547, Cham, 2021. Springer International Publishing. (*) Równy wkład w wykonaną pracę.*

Komentarz: Publikacja została wyróżniona nagrodą na konferencji ICDAR 2021 w kategorii *Best Industry Related Paper Award*.

Wkład autora rozprawy doktorskiej: pomysł użycia architektury BERT-a do problemu ekstrakcji kluczowych informacji i dodaniu do niego cech struktury dokumentu; konceptualizacja i metodologia badania (udział w stałych spotkaniach zespołu); przygotowania zbiorów danych (do trenowania oraz ewaluacji modeli); przygotowanie skryptów do automatyzacji eksperymentów; uruchomienie finalnych eksperymentów na wyzwaniach Kleister NDA, Kleister Charity oraz SROIE; zaproponowanie, zaprojektowanie oraz wykonanie analizy wyników dla sekcji z badaniami ablacyjnymi, edycja publikacji.

2.4.1 Wprowadzenie

Standardowe mechanizmy wykorzystywane w dziedzinie przetwarzania języka naturalnego (w tym ekstrakcji informacji) wczytują tekst jako sekwencję następujących po sobie tokenów (znaków, słów lub części słów). Używając tego podejścia, najnowsze modele oparte o architekturę BERT (przykładowo RoBERTa, Electra, ERNIE) osiągnęły znaczący postęp w rozwiązywaniu większości zadań NLP, zaprezentowanych w benchmarkach GLUE i SuperGLUE [8, 24, 6, 45, 53, 52]. Nie są one jednak dostosowane do przetwarzania dokumentów z bogatą strukturą graficzną, w której do czynienia mamy nie tylko ze zwykłym tekstem, ale również z informacją przedstawioną w formie tabelki lub formularza. W tej pracy zaproponowano mechanizm do wstrzykiwania informacji o strukturze dokumentu, bazując na modelu RoBERTa, który jest poprawioną wersję modelu BERT. Szereg eksperymentów, które przeprowadzono na zbiorach SROIE, CORD, Kleister NDA i Charity, pokazuje istotny wpływ wprowadzonych zmian na skuteczność modelu [13, 30, 43]. Dodatkowo udostępniony został duży zbiór danych do treningu nienadzorowanego, na podstawie którego inni badacze będą w stanie trenować swoje modele w trybie nienadzorowanym.

2.4.2 Zagadnienia powiązane

Reprezentacja tekstu. Współczesne metody w dziedzinie przetwarzania języka naturalnego zamieniają tekst wejściowy, wykonując segmentację na tokeny (reprezentujące znaki, słowa czy segmenty słów) dając w rezultacie ciąg $(t_1, t_2, \ldots, t_\ell)$ o długości ℓ, gdzie tokeny t_i należą do pewnego wcześniej ustalonego skończonego słownika. [33, 1, 8]. Większość z nich opiera się na dwóch mechanizmach dzielących tekst na segmenty słów: BPE (byte pair encoding) oraz model Unigram [41, 21]. Algorytmy te nie są pozbawione wad, które ujawniają się w szczególności, gdy przetwarzamy dokumenty biznesowe [35].

Model BERT [8]. Model BERT opiera się w całości na enkoderze modelu Transformer, wykorzystując do segmentacji algorytm BPE [51]. W tego typu architekturach, każdy token t_i reprezentowany jest przez wektor słowa (ang. *word embedding*) $x_i \in \mathbb{R}^n$, który następnie przekształcany jest przez sieć do wyjściowego wektora liczb $y_i \in \mathbb{R}^m$. Wymiary wektorów wejściowych/wyjściowych wynoszą odpowiednio n oraz m. Wewnątrz takiej sieci znajduje się mechanizm atencji, który nie zakłada z góry ustalonego porządku przetwarzania. Ma to swoje konsekwencje, do których należy zaliczyć:

1. Informacja o pozycji słowa w sekwencji musi być w jakiś sposób uwzględniana w wek-

torze słowa podawanym na wejściu.

2. Wszystkie operacje można wykonywać jednocześnie (w przeciwieństwie do sieci rekurencyjnych).
3. Koszt mnożenia macierzy uzależniony jest głównie od długości sekwencji wejściowej ℓ i wynosi $O(\ell^2 n)$, dlatego długość tekstu na wejściu ograniczana jest najczęściej do 512 tokenów.

Aby uwzględnić pozycję tokenu w sekwencji definiujemy wektor x_i według wzoru:

$$x_i = s_i + p_i, \tag{1}$$

gdzie $s_i \in \mathbb{R}^n$ jest semantycznym wektorem słowa, natomiast $p_i \in \mathbb{R}^n$ jest wektorem pozycji tokenu w sekwencji (nazywanym dalej *wektorem pozycyjnym 1D*). W oryginalnym modelu Transformera wektory pozycji 1D były zakodowane przez stałe wartości dla określonych pozycji, natomiast w modelu BERT te wektory są trenowalne [51, 8]. BERT przed procesem trenowania na docelowym zdaniu wykorzystuje duże zbiory danych do uczenia w trybie nienadzorowanym, tak aby jak najlepiej rozumieć tekst bez konieczności posiadania zbioru trenującego.

Model RoBERTa [24]. Utworzony przez Facebooka model RoBERTa jest następcą modelu BERT (posiada dokładnie taką samą architekturę sieci). Został on jednak zoptymalizowany w taki sposób, żeby osiągać jak najlepsze wyniki na zadaniach docelowych (np. GLUE). Osiągnięto to m.in. poprzez odpowiednią definicję słownika tokenów (segmentów słów) oraz uczenie na dużo większym zbiorze ze zwiększoną liczbą iteracji.

Relatywne kodowanie pozycji. Od momentu pojawienia się architektury Transformera wymyślano coraz to nowsze techniki wstrzykiwania do modelu informacji o pozycji tokenu [9, 20]. Na szczególną uwagę zasługuje metoda relatywnego kodowania pozycji, w której wstrzykujemy informację bezpośrednio do macierzy atencji [42, 37]. W pojedynczej głowie (model podzielony jest na niezależne głowy, które równolegle przetwarzają wejście, a następnie ich wyniki są agregowane) mechanizm atencji transformuje wektory do trzech sekwencji: zapytań (ang. *queries*) $q_i \in \mathbb{R}^d$, kluczy (ang. *keys*) $k_i \in \mathbb{R}^d$ oraz wartości (ang. *values*) $v_i \in \mathbb{R}^d$. Wagi atencji liczone są zgodnie ze wzorem $\alpha_{ij} = d^{-1/2} q_i^T k_j$. Idea relatywnego kodowania polega na modyfikacji wag atencji przez wprowadzenie do wzoru wektora przesunięcia (ang. *bias term*) zgodnie ze wzorem:

$$\alpha'_{ij} = \alpha_{ij} + \beta_{ij}, \tag{2}$$

gdzie parametr $\beta_{ij} = W(i - j)$ jest wyuczalnym wektorem wag uzależnionym od relatywnej pozycji tokenów i oraz j,

2.4.3 Architektura modelu

Schemat architektury modelu LAMBERT zaprezentowany jest na Rysunku 5. Wprowadzone zostały trzy zmiany względem oryginalnego modelu RoBERTa:

1. Modyfikacja wektora wejściowego x_i (wzór 1) o informację o strukturze dokumentu:

$$x_i = s_i + p_i + L(\ell_i). \tag{3}$$

   gdzie parametr $\ell_i \in \mathbb{R}^k$ oznacza wektor pozycyjny 2D opisany szczegółowo w publikacji jako *layout embedding* [11].

2. Modyfikacja wag atencji poprzez dodanie używanego w modelu T5 relatywnego kodowania pozycji [37].

3. Rozszerzenie relatywnego kodowania pozycji (wzór 2) o dwa dodatkowe parametry, związane z relatywną pozycją dwóch tokenów względem odległości od siebie w poziomie oraz w pionie strony:

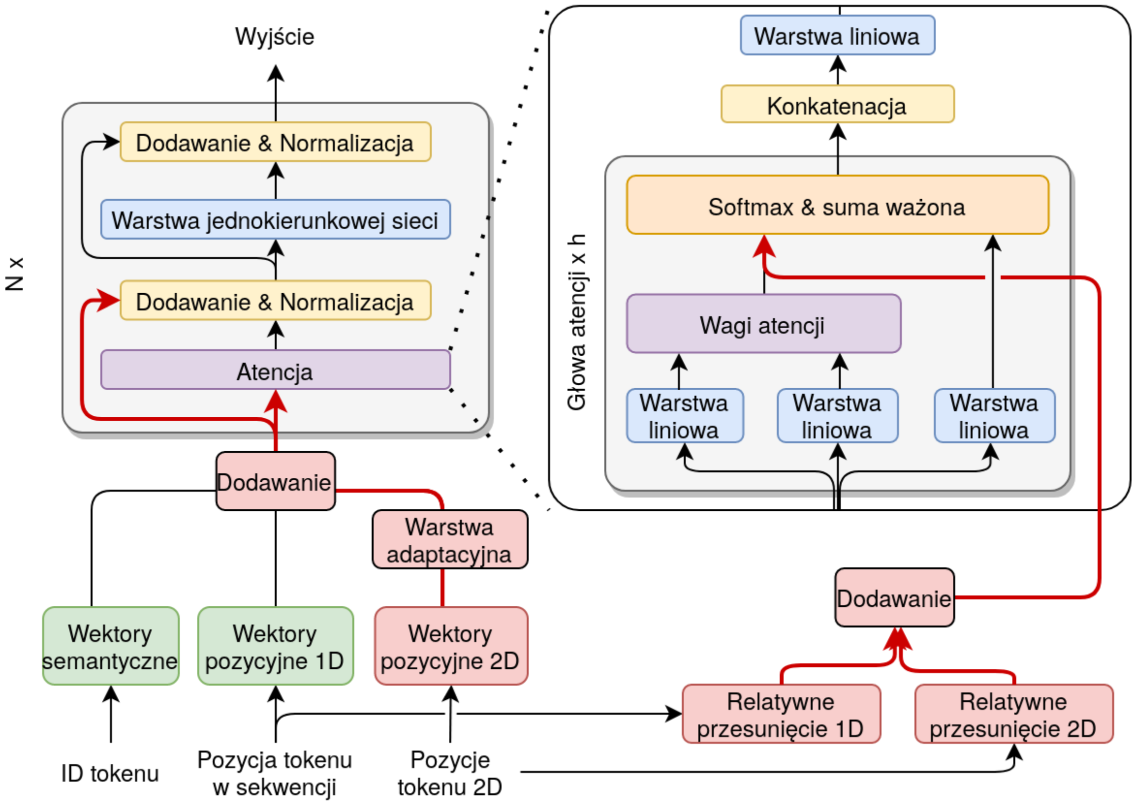
$$\beta_{ij} = W(i - j) + H(\lfloor \xi_i - \xi_j \rfloor) + V(\lfloor \eta_i - \eta_j \rfloor), \tag{4}$$

   gdzie $(\xi_i, \eta_i) = (Cx_1, C(y_1 + y_2)/2)$ są współrzędnymi $i - tego$ tokenu na stronie, wyznaczonymi na podstawie znormalizowanych współrzędnych prostokąta ograniczającego token, $b_i = (x_1, y_1, x_2, y_2)$. $H(\ell)$ i $V(\ell)$ są trenowanymi wagami dla każdej liczby całkowitej z przedziału $\ell \in [-C, C)$ (po serii kilku eksperymentów wartość ta została ustalona na $C = 100$).

2.4.4 Trenowanie modelu w trybie nienadzorowanym

Wprowadzone modyfikacje do oryginalnego modelu RoBERTa wymagały ponownego przetrenowania modelu w trybie nienadzorowanym. Finalny model LAMBERT-a (niektóre modele były trenowane na potrzeby eksperymentów ablacyjnych) został wytrenowany na 8 kartach GPU typu NVIDIA Tesla V100 32GB na specjalnie przefiltrowanym zbiorze plików PDF ściągniętych z wykorzystaniem danych z Common Crawla[9] przez około 1000 iteracji (co odpowiada

[9] https://commoncrawl.org/



Rysunek 5: LAMBERT - architektura modelu. Bloki oznaczone kolorem czerwonym pokazują modyfikacje, które zostały wykonane w stosunku do oryginalnego modelu RoBERTa.

treningowi na 3 milionach stron przez około 25 epok). W celu uzyskania pozycji tokenów na stornie każdy dokument był przetworzony przez wspomniane wcześniej narzędzie do rozpoznawania znaków Tessseract.

2.4.5 Prezentacja i omówienie wyników głównych

W celu weryfikacji jakości modelu LAMBERT przeprowadzona została seria testów na czterech publicznie dostępnych zbiorach danych: Kleister NDA, Charity, SROIE, CORD [43, 13, 30]. Dla zbioru SROIE posiadamy dwa wyniki: jeden wzięty ze strony wyzwania (oznaczony w tabeli jako SROIE), drugi (oznaczony w tabeli jako SROIE*) jest wewnętrznym wariantem, w którym udostępniony przez twórców wyzwania zbiór trenujący został podzielony wewnętrznie na podzbiory trenujący i testowy. Do przeprowadzenia wszystkich eksperymentów wykorzystany był mechanizm opisany w Sekcji 2.2.3 i publikacji [43] z użyciem narzędzia Tesseract do wstępnego przetwarzania dokumentów.

Przeprowadzona seria eksperymentów (Tablica 7) wykazała, że nasz mechanizm znacząco poprawia jakość działania modelu bazowego RoBERTa oraz modelu LayoutLM (zarówno wa-

Model	Liczba parametrów	Nasze eksperymenty				Wyniki zewnętrzne
		NDA	Charity	SROIE*	CORD	SROIE
RoBERTa	125M	77.91	76.36	94.05	91.57	92.39[b]
RoBERTa (16M)	125M	78.50	77.88	94.28	91.98	93.03[b]
LayoutLM	113M	77.50	77.20	94.00	93.82	94.38[a]
	343M	79.14	77.13	96.48	93.62	97.09[b]
LAMBERT (16M)	125M	80.31	79.94	96.24	93.75	—
LAMBERT (75M)	125M	**80.42**	**81.34**	**96.93**	**94.41**	**98.17**[b]

Tablica 7: Porównanie wyników z wykorzystaniem miary F_1 dla przetestowanych modeli. W nawiasach znajduje się liczba stron, na jakich zostały przetrenowane modele. Dla modelu RoBERTa pierwszy wiersz odnosi się do oryginalnego modelu, natomiast drugi odnosi się do modelu przetrenowanego na dokumentach z domeny biznesowo-prawnej. Wyniki [a] na podstawie relewantnej publikacji, [b] na podstawie strony wyzwania SROIE [14]

riantu bazowego, jak i dużego). Istotnym czynnikiem, który rzutuje na końcowe wyniki jest również zbiór danych oraz długość treningu modelu w trybie nienadzorowanym. Zgodnie z panującym trendem w dziedzinie budowania modeli języka im większe i lepsze dane (odfiltrowane z niskiej jakości dokumentów) oraz dłuższy trening, tym model LAMBERT jest skuteczniejszy (patrz model wytrenowany na 16 vs 75 milionach stron) [16].

3 Dorobek naukowy

Sekcja	Publikacja	Punkty MEiN	Cytowania**	Komentarz
2.1	***Tomasz Stanislawek***, *Anna Wróblewska, Alicja Wójcicka, Daniel Ziembicki, and Przemyslaw Biecek. Named Entity Recognition — Is There a Glass Ceiling? In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 624–633, Hong Kong, China, November 2019. Association for Computational Linguistics.*	140	11	
2.2	***Tomasz Stanisławek***, *Filip Graliński, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. Kleister: Key information extraction datasets involving long documents with complex layouts. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, Document Analysis and Recognition – ICDAR 2021, pages 564–579, Cham, 2021. Springer International Publishing.*	140	8	
2.3	*Łukasz Borchmann*, Michał Pietruszka*,* ***Tomasz Stanislawek****, *Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. DUE: End-to-end document understanding benchmark.* `https://openreview.net/forum?id=rNs2FvJGDK`*. 2021.*	0/200	0	(*) Równy wkład w wykonaną pracę. Publikacja jest w trakcie recenzji na konferencję NeurIPS 2021.
2.4	*Łukasz Garncarek*, Rafał Powalski*,* ***Tomasz Stanisławek****, *Bartosz Topolski*, Piotr Halama, Michał Turski, and Filip Graliński. Lambert: Layout-aware language modeling for information extraction. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, Document Analysis and Recognition – ICDAR 2021, pages 532–547, Cham, 2021. Springer International Publishing.*	140	13	(*) Równy wkład w wykonaną pracę. Wyróżniona nagrodą na konferencji ICDAR 2021 w kategorii *Best Industry Related Paper Award.*

Tablica 8: Lista publikacji wchodząca w skład rozprawy doktorskiej. (**) Cytowania wyznaczone na podstawie platformy *https://scholar.google.pl/.*

W Tablicy 8 wyszczególnione zostały wszystkie prace wchodzące w skład rozprawy doktorskiej, w których jestem pierwszym autorem (również tam, gdzie wkład w pracę jest taki sam dla kilku osób). Większość z nich została wygłoszona na cenionych konferencjach naukowych (suma punktów według MEiN wynosi 420, jedna publikacja w trakcie recenzji). Dodatkowo od momentu rozpoczęcia doktoratu uczestniczyłem jeszcze w trzech innych badaniach (Tablica 9) [55, 12, 35], które ściśle wiązały się z tematyką rozprawy. Przed rozpoczęciem studiów doktoranckich pracowałem przy kilku projektach naukowo-badawczych, po realizacji których powstały trzy publikacje [36, 38, 36]. Moje dotychczasowe badania pozwoliły osiągnąć sumaryczną liczbę cytowań 113 oraz h-indeks wynoszący 7.

Publikacja	Cytowania**	Główny wkład w publikację
*Rafal Powalski and **Tomasz Stanislawek**. Unicase–rethinking casing in language models. arXiv preprint arXiv:2010.11936, 2020.*	1	wymyślenie koncepcji, ustalenie metodologii badania oraz pisanie pracy.
*Filip Gralinski, Anna Wróblewska, **Tomasz Stanislawek**, Kamil Grabowski, and Tomasz Górecki. Geval: Tool for debugging NLP datasets and models. In Tal Linzen, Grzegorz Chrupala, Yonatan Belinkov, and Dieuwke Hupkes, editors, Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Blackbo-xNLP@ACL 2019, Florence, Italy, August 1, 2019, pages 254–262. Association for Computational Linguistics, 2019.*	7	Eksperymenty na zadaniach z domeny ekstrakcji informacji, edycja manuskryptu i wymyślanie nowych cech do metody.
*Anna Wróblewska, **Tomasz Stanisławek**, Bartłomiej Prus-Zajączkowski, and Łukasz Garncarek. Robotic process automation of unstructured data with machine learning. In Position Papers of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9-12, 2018, pages 9–16, 2018.*	12	Metodologia badania, pisanie manuskryptu.
*Jarosław Protasiewicz, Witold Pedrycz, Marek Kozłowski, Sławomir Dadas, **Tomasz Stanisławek**, Agata Kopacz, and Małgorzata Gałężewska. A recommender system of reviewers and experts in reviewing problems.Knowledge-Based Systems, 106:164–178, 2016.*	52	Moduł do klasyfikacji publikacji, tworzenie oprogramowania, edycja publikacji.
*Jacek Rapiński, Daniel Zinkiewicz, and **Tomasz Stanislawek**. Influence of human body on radio signal strength indicator readings in indoor positioning systems. Technical Sciences/University of Warmia and Mazury in Olsztyn, (19 (2)):117–127, 2016.*	7	Implementacja i testowanie wybranych algorytmów.
*Jaroslaw Protasiewicz, **Tomasz Stanislawek**, and Slawomir Dadas. Multilingual and hierarchical classification of large datasets of scientific publications. In 2015 IEEE International Conference on Systems, Man, and Cybernetics, pages 1670–1675. IEEE, 2015.*	2	Metodologia badania, implementacja i przeprowadzenie finalnych eksperymentów, pisanie publikacji.

Tablica 9: Lista pozostałych publikacji. (**) Cytowania wyznaczone na podstawie platformy *https://scholar.google.pl/.*

4 Podsumowanie

Badania przeprowadzone w ramach rozprawy doktorskiej i opisane w sekcji 2 dotyczyły ekstrakcji informacji z dokumentów o bogatej strukturze graficznej. W pierwszej kolejności wykonałem analizę jakości aktualnie najlepszych rozwiązań dotyczących przetwarzania zwykłego tekstu (Sekcja 2.1), co pozwoliło mi na identyfikację problemów i możliwych kierunków rozwoju. Następnie skupiłem się na wprowadzeniu do dziedziny nowych zbiorów danych (Sekcja 2.2) uwzględniających liczne problemy, które wcześniej nie były podejmowane i rozwiązywane (m.in. długie dokumenty). Dodatkowo uczestniczyłem w przygotowaniu porównania (Sekcja 2.3), w którym razem z głównymi współautorami przygotowaliśmy zestaw zróżnicowanych zbiorów danych do badania postępu w szerszej dziedzinie, jaką jest rozumienie dokumentów. Ważnym aspektem tego badania była konceptualizacja i wspólny mianownik problemów w całej dziedzinie, a nie tylko ekstrakcji informacji. Za najważniejsze dzieło uznaję prace badawcze związane z powstaniem modelu LAMBERT (Sekcja 2.4). Wspólnie z trzema głównymi członkami zespołu stworzyliśmy nowy model języka, który dodaje informację o pozycji tokenów na stronie, co doprowadziło do uzyskania najlepszych wyników na kilku zbiorach danych: Kleister NDA, Charity i SROIE [11, 14]. Za największy swój wkład tutaj uznaję bycie pomysłodawcą użycia modelu BERT/RoBERTa do problemu ekstrakcji informacji, wspólne przygotowanie wstępnej architektury oraz przeprowadzenie serii eksperymentów ablacyjnych, które doprowadziły do ustalenia najlepszej architektury.

Większość publikacji (jedna praca jest w recenzji na konferencję NIPS 2021) składająca się na niniejszą rozprawę doktorską została wygłoszona na międzynarodowych konferencjach (CoNLL 2019, ICDAR 2021), co pokazuje jakość przeprowadzanych badań. Ich znaczenie podkreśla również fakt zdobycia nagrody w kategorii *Best Industry Related Paper Award* na konferencji ICDAR 2021. Dodatkowo, poza omówionymi w rozprawie publikacjami, byłem jeszcze współautorem (ale nie głównym) trzech prac [55, 12, 35], które częściowo wiążą się z tematyką poruszaną w rozprawie.

4.1 Wkład w rozwój dziedziny

Prace badawcze związane z ekstrakcją informacji z dokumentów o bogatej strukturze graficznej rozpocząłem na początku 2019 roku. Dziedzina ta nie była wówczas szczególnie popularna, a większość rozwiązań tworzona była z wykorzystaniem techniki rozpoznawania obrazów, sieci grafowych lub modeli opartych o sieć z atencją [18, 23, 28]. W tym samym czasie powstały dwa równoległe rozwiązania wykorzystujące po raz pierwszy modele języka: LayoutLM (pierwsze wersja na arxiv pojawiła się 31 grudnia 2019 roku) oraz LAMBERT (publikacja została wysłana na konferencję ACL w połowie grudnia 2019 roku, ale nie została ostatecznie przyjęta w jej pierwszej wersji). Oba te rozwiązania wyznaczyły kierunek badań dla pozostałych naukowców zajmujących się tą tematyką. Dodatkowo przygotowane zbiory danych Kleister stały się ważnym elementem weryfikacji jakości nowo powstałych modeli [56, 3, 50].

4.2 Wdrożeniowe znaczenie przeprowadzonych badań

Prace, które wykonałem podczas studiów doktoranckich, były ściśle związane z aspektem wdrożeniowym i rozwiązywaniem realnych problemów biznesowych w firmie `Applica.ai`. Utworzenie modelu LAMBERT pozwoliło na zwiększenie jakości dostarczanych usług dla już obsługiwanych klientów oraz rozwiązanie dotychczas niemożliwych do realizacji przypadków (np. ekstrakcja danych tabelarycznych). Zbiory danych Kleister pozwoliły na bardziej dokładne przeprowadzenie badań ablacyjnych, co zaowocowało wyborem najlepszego modelu języka. Ponadto, te badania przyczyniły się w dużej mierze do zbudowania następcy modelu LAMBERT, modelu TILT [34].

Słowniczek pojęć

Pojęcia dobrze znane i rozpowszechnione w języku angielskim często nie mają swoich odpowiedników w języku polskim. W związku z czym, aby lepiej zrozumieć przedstawioną pracę, wyróżniłem listę najważniejszych terminów w języku polskim oraz ich tłumaczenie na język angielski. Inspiracje czerpałem głównie z pracy o Narodowym Korpusie Języka Polskiego [49].

Dokumenty o bogatej strukturze graficznej - ang. Visually Rich Documents, VRDs

Rozumienie dokumentów - ang. Document Understanding, DU

Optyczne rozpoznawanie znaków - ang. Optical character recognition, OCR

Tagowanie sekwencyjne - ang. Sequence Labelling

Anotacja - ang. annotation

Ekstrakcja informacji - ang. Information Extraction, IE

Ekstrakcja kluczowych informacji - ang. Key Information Extraction, KIE

Rozpoznawanie jednostek nazewniczych - ang. Named Entity Recognition, NER

Wektor słów/wektor - ang. words embeddings/embedding

Trenować w trybie nienadzorowanym - ang. unsupervised training

Rozumienie obrazów - ang. Computer Vision, CV

Robotyzacja procesów - ang. Robotic Process Automation, RPA

Głowa atencji - ang. Attention head

Literatura

[1] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649. Association for Computational Linguistics, 2018.

[2] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, December 2005.

[3] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. Docformer: End-to-end transformer for document understanding, 2021.

[4] Łukasz Borchmann, Michał Pietruszka, Tomasz Stanislawek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. Due: End-to-end document understanding benchmark. *Openreview*, 2021.

[5] Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. TabFact : A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, April 2020.

[6] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*, 2020.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[8] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[9] Philipp Dufter, Martin Schmitt, and Hinrich Schütze. Position information in transformers: An overview, 2021.

[10] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[11] Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. Lambert: Layout-aware language modeling for information extraction. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, *Document Analysis and Recognition – ICDAR 2021*, pages 532–547, Cham, 2021. Springer International Publishing.

[12] Filip Gralinski, Anna Wróblewska, Tomasz Stanislawek, Kamil Grabowski, and Tomasz Górecki. Geval: Tool for debugging NLP datasets and models. In Tal Linzen, Grzegorz Chrupala, Yonatan Belinkov, and Dieuwke Hupkes, editors, *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@ACL 2019, Florence, Italy, August 1, 2019*, pages 254–262. Association for Computational Linguistics, 2019.

[13] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520, 2019.

[14] ICDAR. Leaderboard of the Information Extraction Task, Robust Reading Competition. `https://rrc.cvc.uab.es/?ch=13&com=evaluation&task=3` (accessed April 7, 2021), 2021.

[15] Daniel Jurafsky and James H. Martin. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., USA, 2009.

[16] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[17] Marcin Kardas, Piotr Czapla, Pontus Stenetorp, Sebastian Ruder, Sebastian Riedel, Ross Taylor, and Robert Stojnic. AxCell: Automatic extraction of results from machine learning papers, 2020.

[18] Anoop R. Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. Chargrid: Towards understanding 2d documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4459–4469. Association for Computational Linguistics, 2018.

[19] Anthony Kay. Tesseract: An open-source optical character recognition engine. *Linux J.*, 2007(159):2, July 2007.

[20] Guolin Ke, Di He, and Tie-Yan Liu. Rethinking positional encoding in language pretraining. *ArXiv*, abs/2006.15595, 2021.

[21] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates, 2018.

[22] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics, 2016.

[23] Xiaojing Liu, Feiyu Gao, Qiong Zhang, and Huasha Zhao. Graph convolution for multimodal information extraction from visually rich documents. *CoRR*, abs/1903.11279, 2019.

[24] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv*, abs/1907.11692, 2019.

[25] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074. Association for Computational Linguistics, 2016.

[26] Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V Jawahar. Infographicvqa, 2021.

[27] Minesh Mathew, Dimosthenis Karatzas, and C.V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2200–2209, January 2021.

[28] Rasmus Berg Palm, Florian Laws, and Ole Winther. Attend, copy, parse end-to-end information extraction from documents. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 329–336, 2019.

[29] Rasmus Berg Palm, Ole Winther, and Florian Laws. Cloudscan - a configuration-free invoice analysis system using recurrent neural networks. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 406–413, 2017.

[30] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. CORD: A Consolidated Receipt Dataset for Post-OCR Parsing. In *Document Intelligence Workshop at Neural Information Processing Systems*, 2019.

[31] Panupong Pasupat and Percy Liang. Compositional semantic parsing on semi-structured tables. *CoRR*, abs/1508.00305, 2015.

[32] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[33] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[34] Rafał Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Pałka. Going full-tilt boogie on document understanding with text-image-layout transformer. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, *Document Analysis and Recognition – ICDAR 2021*, pages 732–747, Cham, 2021. Springer International Publishing.

[35] Rafal Powalski and Tomasz Stanislawek. Unicase–rethinking casing in language models. *arXiv preprint arXiv:2010.11936*, 2020.

[36] Jaroslaw Protasiewicz, Tomasz Stanislawek, and Slawomir Dadas. Multilingual and hierar-

chical classification of large datasets of scientific publications. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1670–1675. IEEE, 2015.

[37] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[38] Jacek Rapiński, Daniel Zinkiewicz, and Tomasz Stanislawek. Influence of human body on radio signal strength indicator readings in indoor positioning systems. *Technical Sciences/University of Warmia and Mazury in Olsztyn*, (19 (2)):117–127, 2016.

[39] Meghan Rimol. Gartner Forecasts Worldwide Hyperautomation-Enabling Software Market to Reach Nearly $600 Billion by 2022. `https://www.gartner.com/en/newsroom/press-releases/2021-04-28-gartner-forecasts-worldwide-hyperautomation-enabling-software-market-`, 2021.

[40] Sebastian Ruder. Challenges and Opportunities in NLP Benchmarking. `http://ruder.io/nlp-benchmarking`, 2021.

[41] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics.

[42] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. 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 464–468, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[43] Tomasz Stanisławek, Filip Graliński, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. Kleister: Key information extraction datasets involving long documents with complex layouts. In Josep Lladós, Daniel Lopresti, and Seiichi Uchida, editors, *Document Analysis and Recognition – ICDAR 2021*, pages 564–579, Cham, 2021. Springer International Publishing.

[44] Nishant Subramani, Alexandre Matton, Malcolm Greaves, and Adrian Lam. A survey of deep learning approaches for ocr and document understanding, 2021.

[45] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 2.0: A continual pre-training framework for language understanding. *arXiv preprint arXiv:1907.12412*, 2019.

[46] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[47] Stacey Svetlichnaya. DeepForm: Understand structured documents at scale. `https://wandb.ai/stacey/deepform_v1/reports/DeepForm-Understand-Structured-Documents-at-Scale--VmlldzoyODQ3Njg`, 2020.

[48] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 2003.

[49] Barbara Lewandowska Tomaszczyk, Mirosław Bańko, Rafał L. Górski, Piotr Pęzik, and Adam Przepiórkowski. Narodowy korpus języka polskiego. 2012.

[50] Benjamin Townsend, Eamon Ito-Fisher, Lily Zhang, and Madison May. Doc2dict: Information extraction as text generation, 2021.

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[52] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*, 2019.

[53] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[54] Mengxi Wei, YIfan He, and Qiong Zhang. Robust layout-aware ie for visually rich documents with pre-trained language models. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 2367–2376, 2020.

[55] Anna Wróblewska, Tomasz Stanisławek, Bartłomiej Prus-Zajączkowski, and Łukasz Garncarek. Robotic process automation of unstructured data with machine learning. In *Position Papers of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9-12, 2018*, pages 9–16, 2018.

[56] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. LayoutLMv2: Multimodal pre-training for visually-rich document understanding, 2020.

[57] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. LayoutLM: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, page 1192–1200, 2020.

[58] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, Lucas Beyer, Olivier Bachem, Michael Tschannen, Marcin Michalski, Olivier Bousquet, Sylvain Gelly, and Neil Houlsby. A large-scale study of representation learning with the visual task adaptation benchmark, 2020.

Deklaracje współautorów

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:
Tomasz Stanislawek, Anna Wróblewska, Alicja Wójcicka, Daniel Ziembicki, and Przemyslaw Biecek. “Named Entity Recognition - Is There a Glass Ceiling?” In: *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 624–633. DOI: 10.18653/v1/K19-1058. URL: https://aclanthology.org/K19-1058
is correctly characterized in the table below.

Contributor	Description of main tasks
Tomasz Stanisławek	– initial idea behind the paper – leading role for the whole process – conceptualization and methodology – implementation and evaluation of selected models – running the experiments – annotation of datasets – results analysis – writing the paper
Anna Wróblewska	– conceptualization and methodology – annotation of datasets – results analysis – writing the paper – supervision in the company to the main author
Alicja Wójcicka	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – writing the paper (annotation process and linguistic categories section) – edition of the manuscript
Daniel Ziembicki	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – edition of the manuscript
Przemysław Biecek	– conceptualization and methodology – results analysis – creating all figures in the paper – edition of the manuscript – supervision to the main author

Tomasz Stanisławek
Tomasz Stanisławek

Anna Wróblewska

Alicja Wójcicka
Alicja Wójcicka

Daniel Ziembicki

Przemysław Biecek
Przemysław Biecek

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:

Tomasz Stanisławek, Anna Wróblewska, Alicja Wójcicka, Daniel Ziembicki, and Przemysław Biecek. "Named Entity Recognition - Is There a Glass Ceiling?" In: *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 624-633. DOI: `10.18653/v1/K19-1058`. URL: `https://aclanthology.org/K19-1058`

is correctly characterized in the table below.

Contributor	Description of main tasks
Tomasz Stanisławek	– initial idea behind the paper – leading role for the whole process – conceptualization and methodology – implementation and evaluation of selected models – running the experiments – annotation of datasets – results analysis – writing the paper
Anna Wróblewska	– conceptualization and methodology – annotation of datasets – results analysis – writing the paper – supervision in the company to the main author
Alicja Wójcicka	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – writing the paper (annotation process and linguistic categories section) – edition of the manuscript
Daniel Ziembicki	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – edition of the manuscript
Przemysław Biecek	– conceptualization and methodology – results analysis – creating all figures in the paper – edition of the manuscript – supervision to the main author

Tomasz Stanisławek

Anna Wróblewska

Anna Wróblewska

Alicja Wójcicka

Daniel Ziembicki

Przemysław Biecek

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:

Tomasz Stanislawek, Anna Wróblewska, Alicja Wójcicka, Daniel Ziembicki, and Przemyslaw Biecek. "Named Entity Recognition - Is There a Glass Ceiling?" In: *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 624–633. DOI: `10.18653/v1/K19-1058`. URL: `https://aclanthology.org/K19-1058`

is correctly characterized in the table below.

Contributor	Description of main tasks
Tomasz Stanisławek	– initial idea behind the paper – leading role for the whole process – conceptualization and methodology – implementation and evaluation of selected models – running the experiments – annotation of datasets – results analysis – writing the paper
Anna Wróblewska	– conceptualization and methodology – annotation of datasets – results analysis – writing the paper – supervision in the company to the main author
Alicja Wójcicka	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – writing the paper (annotation process and linguistic categories section) – edition of the manuscript
Daniel Ziembicki	– conceptualization of model errors in NER task from linguistic perspective – annotation of datasets – edition of the manuscript
Przemysław Biecek	– conceptualization and methodology – results analysis – creating all figures in the paper – edition of the manuscript – supervision to the main author

Tomasz Stanisławek

Anna Wróblewska

Alicja Wójcicka

Alicja Wójcicka

Daniel Ziembicki

Daniel Ziembicki

Przemysław Biecek

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:

Tomasz Stanisławek, Filip Graliński, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. "Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts". In: *Document Analysis and Recognition – ICDAR 2021.* Ed. by Josep Lladós, Daniel Lopresti, and Seiichi Uchida. Cham: Springer International Publishing, 2021, pp. 564–579

is correctly characterized in the table below.

Contributor	Description of main tasks
Tomasz Stanisławek	– conceptualization and methodology – implementation and evaluation of baselines – running the experiments – writing the paper – results analysis – responses to reviewers
Filip Graliński	– conceptualization and methodology – prepare Kleister datasets standard (schema, evaluation tools) – creating Kleister benchmark webpage – edition of the manuscript – results analysis
Anna Wróblewska	– conceptualization and methodology – results analysis – writing the paper – available datasets review – supervision in the company to the main author
Dawid Lipiński	– controlling of the human annotation process – annotation of datasets – writing the paper (annotation process section) – edition of the manuscript
Agnieszka Kaliska	– annotation of datasets – writing the paper (annotation process section) – edition of the manuscript
Paulina Rosalska	– annotation of datasets – writing the paper (annotation process section) – edition of the manuscript
Bartosz Topolski	– implementation of baselines – edition of the manuscript
Przemysław Biecek	– supervision to the main author – edition of the manuscript

Tomasz Stanisławek
Tomasz Stanisławek

Filip Graliński
Filip Graliński

Anna Wróblewska

Dawid Lipiński
Dawid Lipiński

Agnieszka Kaliska

Paulina Rosalska
Rosalska

Bartosz Topolski
B. Topolski

Przemysław Biecek
Przemysław Biecek

September 23, 2021

Declaration

I hereby declare that the contribution to the following paper:
Tomasz Stanisławek, Filip Graliński, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. "Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts". In: *Document Analysis and Recognition – ICDAR 2021*. Ed. by Josep Lladós, Daniel Lopresti, and Seiichi Uchida. Cham: Springer International Publishing, 2021, pp. 564–579
is correctly characterized in the table below (* denotes equal contributions).

Contributor	Description of main tasks
Tomasz Stanisławek	– conceptualization and methodology
	– implementation and evaluation of baselines
	– running the experiments
	– writing the paper
	– results analysis
	– responses to reviewers
Filip Graliński	– conceptualization and methodology
	– prepare Kleister datasets standard (schema, evaluation tools)
	– creating Kleister benchmark webpage
	– edition of the manuscript
	– results analysis
Anna Wróblewska	– conceptualization and methodology
	– results analysis
	– writing the paper
	– available datasets review
	– supervision in the company to the main author
Dawid Lipiński	– controlling of the human annotation process
	– annotation of datasets
	– writing the paper (annotation process section)
	– edition of the manuscript
Agnieszka Kaliska*	– annotation of datasets
	– writing the paper (annotation process section)
	– edition of the manuscript
Paulina Rosalska*	– annotation of datasets
	– writing the paper (annotation process section)
	– edition of the manuscript
Bartosz Topolski	– implementation of baselines
	– edition of the manuscript
Przemysław Biecek	– supervision to the main author
	– edition of the manuscript

Tomasz Stanisławek

Filip Graliński
Filip Gralinski

Anna Wróblewska
Anna Wróblewska

Dawid Lipiński

Agnieszka Kaliska

Paulina Rosalska

Bartosz Topolski

Przemysław Biecek

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:
Łukasz Borchmann, Michał Pietruszka, Tomasz Stanisławek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. "DUE: End-to-End Document Understanding Benchmark". In: *Under review in NeurIPS 2021*. 2021. URL: `https://openreview.net/forum?id=rNs2FvJGDK`
is correctly characterized in the table below (* denotes equal contributions).

Contributor	Description of main tasks
Łukasz Borchmann*	– conceptualization and methodology (participated in regular discussions) – methodology of the considered datasets for DUE benchmark – implementation of baselines – create DUE benchmark webpage – create scripts for evaluation – convert documents from TabFact and WTQ datasets into pdf files – result analysis – writing the paper – organizing and controlling the process of human annotation
Michał Pietruszka*	– conceptualization and methodology (participated in regular discussions) – methodology and preparation list of the considered datasets for DUE benchmark – implementation of baselines – preparation of datasets (DocVQA, InfographicsVQA, WikiTableQuestions, PWC) – preparing code, models and datasets for final release – result analysis – writing the paper – organizing and controlling the process of human annotation
Tomasz Stanisławek*	– conceptualization and methodology (participated in regular discussions) – methodology and preparation list of the considered datasets for DUE benchmark – prepare schema for storing benchmark datasets in unified data format – preparation of datasets (Kleister Charity, DeepForm, TabFact) – curation of PWC and DeepForm datasets – methodology for creation of the diagnostic subsets – result analysis – improved the first version of the paper / edition of the manuscript – organizing and controlling the process of human annotation
Dawid Jurkiewicz	– participated in regular discussions – implementation of baselines – significantly improved results of the baselines (hyper-param search for it) – performing experiments – preparing code and models for final release – edition of the paper
Michał Turski	– methodology and preparation of the diagnostic subsets – organizing and controlling the process of human annotation – controlling the process of measuring human performance where it was required (PWC, DeepForm, WTQ) – edition of the paper
Karolina Szyndler	– improving schema for storing benchmark datasets in unified data format – code for reading datasets by the baselines
Filip Graliński	– participated in regular discussions – edition of the paper

Łukasz Borchmann	Michał Pietruszka	Tomasz Stanisławek	Dawid Jurkiewicz
Michał Turski	Karolina Szyndler	Filip Graliński	
Michał Turski	Karolina Szyndler	Filip Graliński	

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:
Łukasz Borchmann, Michał Pietruszka, Tomasz Stanisławek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. "DUE: End-to-End Document Understanding Benchmark". In: *Under review in NeurIPS 2021.* 2021. URL: `https://openreview.net/forum?id=rNs2FvJGDK`
is correctly characterized in the table below (* denotes equal contributions).

Contributor	Description of main tasks
Łukasz Borchmann*	– conceptualization and methodology (participated in regular discussions) – methodology of the considered datasets for DUE benchmark – implementation of baselines – create DUE benchmark webpage – create scripts for evaluation – convert documents from TabFact and WTQ datasets into pdf files – result analysis – writing the paper – organizing and controlling the process of human annotation
Michał Pietruszka*	– conceptualization and methodology (participated in regular discussions) – methodology and preparation list of the considered datasets for DUE benchmark – implementation of baselines – preparation of datasets (DocVQA, InfographicsVQA, WikiTableQuestions, PWC) – preparing code, models and datasets for final release – result analysis – writing the paper – organizing and controlling the process of human annotation
Tomasz Stanisławek*	– conceptualization and methodology (participated in regular discussions) – methodology and preparation list of the considered datasets for DUE benchmark – prepare schema for storing benchmark datasets in unified data format – preparation of datasets (Kleister Charity, DeepForm, TabFact) – curation of PWC and DeepForm datasets – methodology for creation of the diagnostic subsets – result analysis – improved the first version of the paper / edition of the manuscript – organizing and controlling the process of human annotation
Dawid Jurkiewicz	– participated in regular discussions – implementation of baselines – significantly improved results of the baselines (hyper-param search for it) – performing experiments – preparing code and models for final release – edition of the paper
Michał Turski	– methodology and preparation of the diagnostic subsets – organizing and controlling the process of human annotation – controlling the process of measuring human performance where it was required (PWC, DeepForm, WTQ) – edition of the paper
Karolina Szyndler	– improving schema for storing benchmark datasets in unified data format – code for reading datasets by the baselines
Filip Graliński	– participated in regular discussions – edition of the paper

Łukasz Borchmann Michał Pietruszka Tomasz Stanisławek Dawid Jurkiewicz

Michał Pietruszka Tomasz Stanisławek Dawid Jurkiewicz

Michał Turski Karolina Szyndler Filip Graliński

September 24, 2021

Declaration

I hereby declare that the contribution to the following paper:
Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. "LAMBERT: Layout-Aware Language Modeling for Information Extraction". In: *Document Analysis and Recognition – ICDAR 2021.* Cham: Springer International Publishing, 2021, pp. 532–547
is correctly characterized in the table below (* denotes equal contributions).

Contributor	Description of main tasks
Łukasz Garncarek*	– conceptualization and methodology (participated in regular team discussions) – implemented basic framework for models with context embeddings – result analysis for ablation study section – wrote the paper
Rafał Powalski*	– conceptualization and methodology (participated in regular team discussions) – implemented basic framework for models with context embeddings – proposed and implemented relative embeddings – prepared SROIE dataset and run initial experiments – edition of the paper
Tomasz Stanisławek*	– initialize the whole idea of using BERT based model to Key Information Extraction task – conceptualization and methodology (participated in regular team discussions) – prepared datasets (for model training and model evaluation of downstream tasks) – implemented scripts for automation of experiments on downstream tasks – run final experiments on NDA, Kleister and SROIE datasets – propose, design and do result analysis for the ablation study section – edition of the paper
Bartosz Topolski*	– conceptualization and methodology (participated in regular team discussions) – implemented scripts for automation of experiments on downstream tasks – run initial experiments on downstream tasks – significantly improved framework for models with context embeddings – edition of the paper
Piotr Halama	– prepare scripts for running experiments on LayoutLM – run all experiments for CORD dataset – edition of the paper
Michał Turski	– create final dataset for model training with methodology for filtering out non business/legal documents – implementation of training flow for training LAMBERT models – edition of the paper
Filip Graliński	– supervised team (participated in regular team discussions) – evaluation methodology – review of the paper

Łukasz Garncarek

Rafał Powalski
(podpis) Rafał Powalski

Tomasz Stanisławek
Tomasz Stanisławek

Bartosz Topolski
B. Topolski

Piotr Halama
Piotr Halama

Michał Turski
Michał Turski

Filip Graliński
Filip Graliński

Appendix A: Named Entity Recognition - Is there a glass ceiling?

Named Entity Recognition - Is there a glass ceiling?

Tomasz Stanislawek[†,‡], **Anna Wróblewska**[†,‡], **Alicja Wójcika**[†,§],
Daniel Ziembicki[†,§] **Przemyslaw Biecek**[‡,¶]

[†]Applica.ai, Warsaw, Poland
[¶]Samsung Research Poland, Warsaw, Poland
[‡]Faculty of Mathematics and Information Science, Warsaw University of Technology
[§]Department of Formal Linguistics, University of Warsaw

Abstract

Recent developments in Named Entity Recognition (NER) have resulted in better and better models. However, is there a glass ceiling? Do we know which types of errors are still hard or even impossible to correct? In this paper, we present a detailed analysis of the types of errors in state-of-the-art machine learning (ML) methods. Our study reveals the weak and strong points of the Stanford, CMU, FLAIR, ELMO and BERT models, as well as their shared limitations. We also introduce new techniques for improving annotation, for training processes and for checking a model's quality and stability.

Presented results are based on the CoNLL 2003 data set for the English language. A new enriched semantic annotation of errors for this data set and new diagnostic data sets are attached in the supplementary materials.

1 Introduction

The problem of Named Entity Recognition (NER) was defined over 20 years ago at the Message Understanding Conference (MUC, 1995; Sundheim, 1995). Nowadays, there are a lot of solutions capable of a very high accuracy even on very hard and multi-domain data sets (Yadav and Bethard, 2018; Li et al., 2018).

Many of these solutions benefit from large available data sets or from recent developments in deep neural networks. However, in order to progress further with this last mile, we need a better understanding of the sources of errors in NER problem; as it is stated that *”The first step to address any problem is to understand it”*. We performed a detailed analysis of errors on the popular CoNLL 2003 data set (Tjong Kim Sang and De Meulder, 2003).

Of course, different models make different mistakes. Here, we have focused on models that constitute a kind of breakthrough in the NER domain. These models are: Stanford NER (Finkel et al., 2005), the model made by the NLP team from Carnegie Mellon University (CMU) (Lample et al., 2016), ELMO (Peters et al., 2018), FLAIR (Akbik et al., 2018) and BERT-Base (Devlin et al., 2018). In the Stanford model, Conditional Random Fields (CRF) with manually created features were tackled. Lample and the team (at CMU) used an LSTM deep neural network with an output with CRF for the first time. ELMO and FLAIR are new language modeling techniques as an encoder, and LSTM with a CRF layer as an output decoder. A team from Google used a fine-tuning approach with the BERT model in a NER problem for the first time, based on a Bi-diREctional Transformer language model (LM).

We analyzed the data set from a linguistic point of view in order to understand problems at a deeper level. As far as we know only a few studies analyse in details errors for NER problems (Niklaus et al., 2018; Abudukelimu et al., 2018; Ichihara et al., 2015). They mainly explore a range of name entities (boundaries in a text) and the precision and popular metrics of a class prediction (precision, recall, F1). We found the following discussions valuable:

- (Abudukelimu et al., 2018) on annotation and extraction of Named Entities,
- (Braşoveanu et al., 2018) on an analysis of errors in Named Entity Linking systems,
- (Manning, 2011) on linguistic limitations in building a perfect Part-of-Speech Tagger.

We took a different approach. First, our team of data scientists and linguists defined 4 major and

Proceedings of the 23rd Conference on Computational Natural Language Learning, pages 624–633
Hong Kong, China, November 3-4, 2019. ©2019 Association for Computational Linguistics

11 minor categories of types of problems typical for NLP (see Tab. 2). Next, we acquired all erroneous samples (containing errors in model outputs) and we assigned them to the newly defined categories. Finally, we characterized the incorrect output of the models with regard to gold standard annotations and following our team's consensus.

Accordingly, our overall contribution is a conceptualization and classification of the roots of problems with NER models as well as their characterization. Moreover, we have prepared new diagnostic sets for some of our categories so that other researchers can check the weakest points of their NER models.

In the following sections, we introduce our approach regarding the re-annotation process and model evaluation (section 2); we also show and discuss the results (section 3). Finally, we conclude our paper with a discussion (section 4) and draw conclusions (section 5).

2 Method

We commenced our research by reproducing the selected models for the CoNLL 2003 data set[1]. Then, we analysed the erroneous samples, sentences from the test set. It is worth mentioning that we analysed the most common types of named entities, i.e. PER - names of persons, LOC - location names, ORG - organization names. Having several times reviewed the model results and the error-prone data set, we defined the linguistic categories that are the most probable sources of model mistakes. As a result, we were able to annotate the samples with these categories; we then analysed the results and found a few possible improvements.

2.1 Models description

A brief history of the key developments of NER models for the CoNLL data is listed in Table 1. In our analysis, we chose 5 models (bold in the table) that make up significant progress.

Stanford NER CRF was the first industry-wide library to recognize NERs (Finkel et al., 2005). The LSTM layer put forward by Lample from Carnegie Mellon University (CMU) was the first deep learning architecture with a CRF output layer (Lample et al., 2016). The following: a token-based language model (LM) with bi-LSTM with CRF (ELMO) (Peters et al., 2018), a character-based LM with the same output (FLAIR) (Akbik et al., 2018) and a bi-directional language model based on an encoder block from the transformer architecture (BERT) with a fine tune classification output layer (Devlin et al., 2018) are very important techniques; and that not only in the domain of NER.

Model	F1
Ensemble of HMM, TBL, MaxEnt, RRM (Florian et al., 2003)	88.76
Semi-supervised learning (Ando and Zhang, 2005)	89.31
Stanford CRF (Finkel et al., 2005)	87.94
Neural network (Collobert et al., 2011)	89.59
CRF & lexicon embeddings (Passos et al., 2014)	90.90
CMU LSTM-CRF (Lample et al., 2016)	90.94
Bi-LSTM-CNNs-CRF (Ma and Hovy, 2016)	91.21
ELMO: Token based LM Bi-LSTM-CRF (Peters et al., 2018)	92.22
BERT-base: Fine tune Bi-Transformer LM with BPE token encoding (Devlin et al., 2018)	92.4 (*)
CVT: Cross-view training with Bi-LSTM-CRF (Clark et al., 2018)	92.61
BERT-large: Fine tune Bi-Transformer LM with BPE token encoding (Devlin et al., 2018)	92.8 (*)
FLAIR: Char based LM + Glove with Bi-LSTM-CRF (Akbik et al., 2018)	93.09 (**)
Fine tune Bi-Transformer LM with CNN token encoding (Baevski et al., 2019)	93.5

Table 1: Results reported in authors' publications about NER models on the original CoNLL 2003 test set. (*) There is no script for replicating these results and also hyper-parameters were not given. See a discussion at (google bert, 2019) (**) This result was not achieved with the current version of the library. See a discussion at (Flair, 2018) and the reported results at (Akbik et al., 2019)

2.2 Linguistic categories

From a human perspective, the task of NER involves several sources of knowledge: the situation in which the utterance was made, the context of

[1] The details of the model parameters are described in our supplementary materials.

other texts and utterances in the particular domain, the structure of the sentence, the meaning of the sentence, and general knowledge about the world.

While designing categories for annotation, we tried to define these layers of NEs understanding; however, some of them are particularly problematic. For example, there is a problem with a distinction between the meaning (of lexical items and of a whole sentence) and general knowledge. Since there is an enormous and relentless linguistic and philosophical debate on this topic (Rey, 2018), we decided not to delimit these categories and not to distinguish them. Therefore, they have been labeled together as 'sentence level context' (SL-C).

Consequently, we ended up with a set of categories for annotating the items (sentences) from our data set, which are presented in Table 2 as well as described briefly in the following sections and more precisely in the supplementary materials. We have also added more examples for each category in this material.

shortcut	linguistic property
DE-	Data set Errors
DE-A	Annotation errors
DE-WT	Word Typos
DE-BS	Word/Sentence Bad Segmentation
SL-	Sentence Level dependency
SL-S	Sentence Level Structure
SL-C	Sentence Level Context
DL-	Document Level dependency
DL-CR	Document Co-Reference
DL-S	Document Structure
DL-C	Document Context
G-	General properties
G-A	General Ambiguity
G-HC	General Hard Case
G-I	General Inconsistency

Table 2: Linguistic categories prepared for our annotation procedure.

DE-A: Annotation errors are obvious errors in the preliminary annotations (the gold standard in the CoNLL test data set). For example: in the sentence *"SOCCER - JAPAN GET LUCKY WIN, CHINA IN SURPRISE DEFEAT"* as a gold standard annotation *"CHINA"* is assigned a person type; it should, however, be defined as a location so as to be consistent with the other sentence annotations.

DE-WT: Word typos are simple typos in any word in a sample sentence, for exmple: *"Pollish"* instead of *"Polish"*.

DE-BS: Word-sentence bad segmentation. We annotated this case if a few words, joined together with a hyphen or separated by a space, were incorrectly divided into tokens (e.g. *"India-South"*), or where a sentence was erroneously divided inside a boundary of a named entity, which prevented its correct interpretation. For example: in the data set there is a sentence divided into two parts: *"Results of National Hockey"* and *"League"*.

SL-S: Sentence level structure dependency occurs when there is a special construction within a sentence (a syntactic linguistic property) that is a strong premise for defining an entity. In the studied material, we distinguished two such constructions: brackets and bullets. The error receives the SL-S annotation, when the system should have been able to recognize a syntactic linguistic property that leads to correct NER tagging but failed to do so and made a NER mistake. For example: one of the analysed NER systems did recognize all locations except *"Philippines"* in the following enumerating sentence: *"ASEAN groups Brunei, Indonesia, Malaysia, the Philippines, Singapore, Thailand and Vietnam."*.

SL-C: Sentence level context cases are those in which one is able to define an appropriate category of NE based only on the sentence context. For example: one of NER systems has a problem with recognizing the organization *"Office of Fair Trading"* in the sentence: *"Lang said he supported conditions proposed by Britain's Office of Fair Trading, which was asked to examine the case last month."*.

DL-CR: Document level co-reference category was annotated if there was a reference within a sentence to an object that was also referred to in another sentence in the same document. For example: evaluating the *"Zywiec"* named entity in the sentence *"Van Boxmeer said Zywiec had its eye on Okocim ..."*, it has to be considered that there is another sentence in the same document in the data set that explains the organization name, which is: *"Polish brewer Zywiec's 1996 profit..."*.

DL-S: Document level structure cases are those in which the structure of a document plays an important role, i.e. the occurrence of objects in the table (for example the headings determine the scope of an entity itself and its category). For example: look at the following three sentences, which obviously compose a table: *”Port Loading Waiting”*; *”Vancouver 5 7”*, *”Prince Rupert 1 3”*. One of our NER systems had a problem with recognizing each localisation inside the table; however, the system recognized the header as a named entity.

DL-C: Document level context is a type of a linguistic category in which the entire context of a document (containing an annotated sentence) is needed in order to determine a category of an analysed entity, and in which none of the sentence level linguistic categories has been assigned (neither SL-S and SL-C).

G-A: General ambiguity are those situations in which an entity occurs in a different sense from that in which this word (entity) is used in its most common understanding and usage. For example: the common word *’pace’* may as well be occur to be a surname, as in the following sentence: *”Pace, a junior, helped Ohio State...”*.

G-HC: General hard cases are cases occurring for the first time in a set in a given subtype, and which can be interpreted in two different ways. For example: *”Real Madrid’s Balkan strike force...”* where the word *’Balkan’* can be a localisation or an adjective.

G-I: General inconsistency are cases of inconsistencies in the annotation (in the test set itself as well as between the training and test sets). For example in the sentence: *”... Finance Minister Eduardo Aninat said.”*, the word *’Finance’* is annotated as an organisation but in the whole data set the names of ministries are not annotated in the context of the role of a person.

2.3 Annotation procedure

All those entities that had been incorrectly recognized by any of the tested modelsfalse positives, false negatives and wrongly tagged entities were annotated in our research by two teams. Each team consisted of a linguist and a data scientist. We did not analyse errors with the MISC entity type, but the person, localisation and organisation names. The MISC type comprises a variety of NERs that are not of other types. Its definition is rather vague and it is hard to conceptualize what it actually means, e.g. if whether it comprises events or proper names, or even adjectives.

The annotation process was performed in four steps:

1. a set of linguistic annotation categories was established, see the previous section 2.2;
2. the data set was split into two equal parts: one part for each team; all entities were annotated twice, by a linguist and by a data scientist, each working independently;
3. the annotations were compared and all inconsistencies were solved within each team;
4. two teams checked the consistency of the other team’s annotations; all borderline and dubious cases were discussed by all team members and reconciled.

The inter-annotator agreement statistics and Kappa are presented in Table 3. A few categories were very difficult to conceptualize, so it took more time to solve these inconsistencies. In these inconsistent cases, two annotators (a linguist and a data scientist) thoroughly discussed each example.

Not all categories (see Table 2) were annotated by the whole team. Those easy to annotate, as the categories regarding simple errors (i.e. DE-A, DE-WT, DE-BS), were done by one person and then just checked by another.

The general inconsistencies category (G-I) were done semi-automatically and then checked. The semi-automatic procedure was as follows: first finding similarly named entities in the training and test sets and then looking at their labels. By ’similarly named entities’ we mean, e.g. a division of an organization having a geographical location in its name (”Pacific Division”), or a designation of a person from any country (”Czech ambassador”).

Additionally, a document level context (DL-C) category was derived from the rule of not being present in any sentence level category (i.e. SL-C or SL-S).

2.4 Our diagnostic procedure

The next step, after the analysis of linguistic categories of errors, was to create additional diagnostic sets. The goal of this approach was to find, or create, more examples that reflect the most challenging linguistic properties; these can be sentence and document level dependencies and can also include a few ambiguous examples. These ambiguities are for instance names that contain words in common usage. We selected 65 examples

annotated class	agreement [%]	Kappa
SL-S	94.99	0.572
SL-C	69.64	0.389
DL-CR	78.00	0.554
DL-S	81.44	0.536
G-A	68.96	0.252
G-HC	74.46	0.340

Table 3: Inter-annotator statistics (agreement and Kappa) at the very first stage of the annotation procedure, before discussing each controversial example and the super-annotation stage. The statistics are calculated for those categories that were annotated by human annotators.

from Wikipedia articles per two groups of linguistic problems: sentence-level and document-level contexts.[2]

The first diagnostic set comprises sentences in which the properties of a language, general knowledge or a sentence structure are sufficient to identify a NE class. We use this Template Sentences (TS) to check whether a model will have the same quality after changing words, i.e. a name of an entity. For each sentence we prepared at least 2 extra entities with different lengths of words which are well suited to the context. For example in a sentence: *"Atlético's best years coincided with dominant Real Madrid teams."*, the football team *"Atlético"* can be replaced with *"Deportivo La Coruña"*.

The second batch of documents was a group of sentences in which a sentence context is not sufficient to designate a NE, so we need to know more about the particular NE, e.g. we need to look for its co-references in the document, or we require more context, e.g. a whole table of sports results, not only one row. (This particular case often occurs in the CoNLL 2003 set when referring to sports results.) We called this data set Document Context Sentences (DCS). In this data set we annotated NEs and their co-references that are also NEs. An example of such a sentence and its context is as follows: *"In 2003, Loyola Academy (X, ORG) opened a new 60-acre campus ... The property, once part of the decommissioned NAS Glenview, was purchased by Loyola (X,ORG) in 2001."* The second occurrence of the *"Loyola"* name is difficult to recognize as an organization without its first occurrence, i.e. *"Loyola Academy"*.

The other type of a diagnostic set is fairly simple. It is generated from random words and letters that are capitalized or not. Its purpose is just to check if a model over-fits a particular data set (in our case, the CoNLL 2003 set). A scrutinized model should not return any entities on those Random Sentences (RS). We generated 2 thousands of these pseudo-sentences.

3 Results

3.1 Annotation quality

In Table 4 we gathered our model's results for the standard CoNLL 2003 test set and the same set after the re-annotation and correction of annotation errors. We replaced only those annotations (gold standard) which we (all team members) were sure of. Those sentences in which the class of an entity occurrence was ambiguous were not corrected. This shows that the models are better than we thought they were, and so we corrected only the test set and left the inconsistencies.[3].

	Stanford	CMU	ELMO	FLAIR	BERT
ALL-O	88.13	89.78	92.39	92.83	91.62
ALL-C	88.73	90.39	93.21	**93.79**	92.33
PER-O	93.31	95.74	97.07	97.49	96.14
PER-C	93.94	96.49	97.81	**98.08**	96.88
ORG-O	84.23	86.90	90.68	91.34	90.61
ORG-C	84.89	87.53	91.61	**92.64**	91.44
LOC-O	90.83	92.02	93.87	94.01	92.85
LOC-C	91.58	92.62	**94.92**	94.72	93.59
MISC-O	79.10	77.31	82.31	82.89	80.81
MISC-C	79.37	77.58	82.47	**84.40**	81.10

Table 4: Results for selected models on the original (designated as ending '...-O') and re-annotated / corrected ('...-C') CoNLL 2003 test set concerning NE classes (ALL comprise PER, ORG, LOC, MISC). The given metric is a multilabel-F1 score (percentages).

3.2 Linguistic categories statistics

In the CoNLL 2003 test set, we chose as samples words and sentences in which at least one model made a mistake. The set of errors comprises 1101

[2] Our prepared diagnostic data sets are available at https://github.com/applicaai/ner-resources

[3] A small part of the data set of annotation corrections and also the debatable cases will be available at our github – https://github.com/applicaai/ner-resources. We decided not to open the whole data set, because it is the test set and the tuning models on this set would lead to unfair results. On the other hand, we could not perform the analysis on a validation set because it is rather poor with respect to different kinds of linguistic properties.

named entities. The results of each model on this set in terms of our linguistic categories are presented in Fig. 1, Fig. 2 and in Table 5.

Most mistakes were made by the Stanford and CMU models, 703 and 554 respectively. ELMO, FLAIR and BERT, which use contextualised language models, performed much better. These embedded features help the models to understand words in their context and thus resolve most problems with ambiguities.

The CMU model has most problems with sentence level context and ambiguity. This is probably due to the fact that this model uses non-contextualized embedded features (Fig. 2). The Stanford model fares the worst in terms of structured data (almost twice as many errors as the other models), which means that it is not good at defining an entity type within a very limited context (Tab. 5). The Stanford model's hand-crafted features do not store information about the probabilities of words which could represent a specific entity type. It generates much more errors than the other models.

	Stanford	CMU	ELMO	FLAIR	BERT
DE-WT	10	6	9	8	10
DE-BS	38	39	33	33	40
SL-S	46	21	13	16	11
SL-C	448	378	250	223	300
DL-CR	372	316	198	184	263
DL-S	202	107	97	100	117
DL-C	247	175	144	146	170
G-A	219	183	98	101	94
G-HC	72	68	65	59	65
G-I	19	20	21	20	20
Errors	703	554	395	370	472
Unique errors	235	93	23	12	79

Table 5: Number of errors for a particular model and a particular class of errors. The total number of annotated errors is 1101.

Modern techniques using contextualized language models like ELMO, FLAIR and BERT reduced a number of mistakes in SL-C category by more than 50% in comparison to the Stanford model. But they are unable to fix most errors in general problems related to inconsistency (G-I), general hard cases (G-HC) or word typos (DE-WT). See Figure 4 for more details.

Nevertheless, there are still a lot of common problems (27.8%). In common errors (Fig. 3), SL-C (sentence level context) and DL-CR (document level co-reference) co-occur the most often. Thus, if a model also takes into account the context of a whole document, it can be of great benefit. Considering a document structure (DL-S) in modeling is also very important. This also can help to resolve a lot of ambiguity issues (G-A). Here is an example of such a situation: *"Pace outdistanced three senior finalists..."*, *"Pace"* is a person's surname, but one is able to find it out only when analysing the whole document and finding references to it in other sentences that directly point to the class of the named entity.

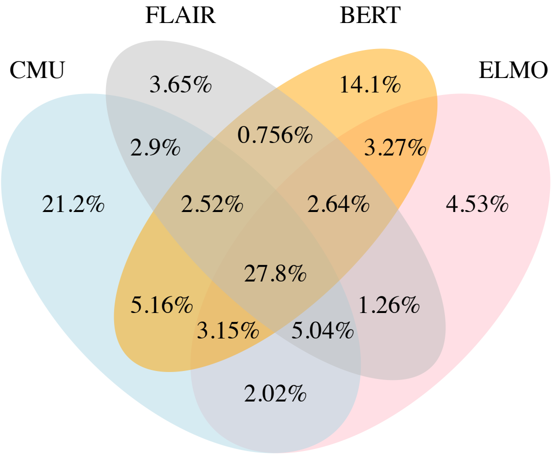


Figure 1: Venn diagram for errors in the CMU, FLAIR, BERT, ELMO models. The four models generate 794 errors and 221 are common to all of them. The Stanford model as the most error-prone is here not referred to.

We must be aware of the fact that some problems cannot be resolved with this data set, not even in general. Those problems have roots in two main areas: data set annotation (word typos, bad segmentation, inconsistencies) and a complicated structure of a language. Generally in most languages it is easier to say what entity represents a real word instance than to define an exact entity type (especially when we use a metonymic sense of a word), e.g. 'Japan' can be a name of a country or of a sports team.

3.3 Diagnostic data sets

Looking at the models' results in our diagnostic data sets (Tab. 6), the first and most important observation is that we achieved significantly lower results than originally on the CoNLL 2003 test

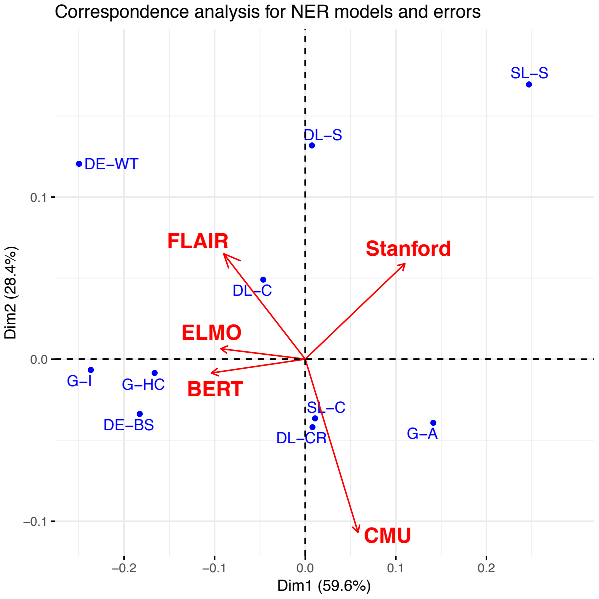
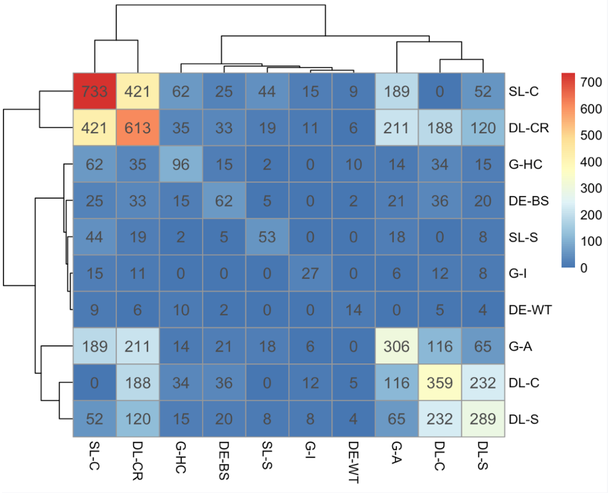


Figure 2: Correspondence analysis for the models' errors. ELMO, FLAIR and BERT are more affected by G-HC and G-I, FLAIR is also reduced with DL-C and DE-WT. See Table 5 for more details and Table 2 for names of categories.



	SL-C	DL-CR	G-HC	DE-BS	SL-S	G-I	DE-WT	G-A	DL-C	DL-S
SL-C	733	421	62	25	44	15	9	189	0	52
DL-CR	421	613	35	33	19	11	6	211	188	120
G-HC	62	35	96	15	2	0	10	14	34	15
DE-BS	25	33	15	62	5	0	2	21	36	20
SL-S	44	19	2	5	53	0	0	18	0	8
G-I	15	11	0	0	0	27	0	6	12	8
DE-WT	9	6	10	2	0	0	14	0	5	4
G-A	189	211	14	21	18	6	0	306	116	65
DL-C	0	188	34	36	0	12	5	116	359	232
DL-S	52	120	15	20	8	8	4	65	232	289

Figure 3: Heatmap for errors from the five considered models. 197 errors are common to all the models. In this figure we can see which linguistic categories tend to occur together.

set[4]. The reason for this is that diagnostic examples were selected for a broader range of topics (not only politics or sports). In particular, document context sentences (DCS) contain 364 unique entities of which only 47 appeared in an exact word form in the training data, and only 42 of them have the same entity type (organization, location or person) - the same type as in the CoNLL 2003 training set. Additionally, those sentences are also difficult due to their linguistic properties (for some entities you must analyse a whole article to properly distinguish their type).

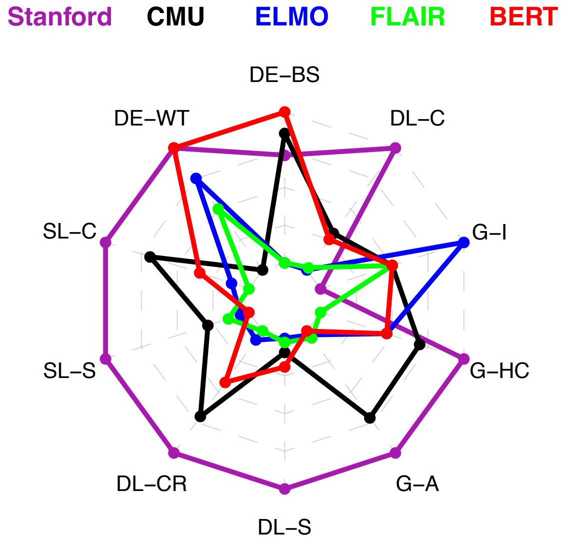


Figure 4: Radar plot with the strong and weak sides of NER models. A radius corresponds to a number of errors in a given linguistic category, the smaller the better. See Table 5 for more details.

As far as the results of the diagnostic sets are concerned, we observed much better results for solutions using embeddings generated by the language models. It seems that by using ELMO embeddings we can outperform the FLAIR and BERT-Base models in case of sentences about general topics, in which the context of a whole sentence is more important than properties of words composing entities.

Moreover, when we tested all the models on random sentences (RS), this was not so good as we might have expected. All the models are very sensitive to words starting with or consisting of capital letters. Results from this diagnostic set could help to choose a model that must work properly on documents which were produced by the OCR engine with their many mistakes and misspellings.

Another interesting idea is to train or just test a model on some template sentences (TS). With such a data set we can test a model's ability to detect proper boundaries of an entity. We can do it by replacing a template entity with another one consisting of a different number of words. We could also adjust our models to a particular domain, e.g.to change entities with a PERSON type in an original data set to be more globally diversified, if we have to extract person names from the

[4] We add statistics and a few examples from our diagnostic data sets in the supplementary materials.

whole world (Asian or Russian names).

	Stanford	CMU	ELMO	FLAIR	BERT
DCS (F1)	45.37	61.86	**76.36**	71.89	68.90
DCS (P)	43.66	58.07	73.11	69.35	59.06
DCS (R)	47.21	66.17	79.92	74.63	82.66
TS-O (F1)	68.96	79.66	**89.45**	88.51	83.47
TS-O (P)	76.92	78.33	85.48	85.25	75.18
TS-O (R)	62.50	81.03	93.81	92.04	93.81
TS-R (F1)	63.06	72.86	85.01	**86.63**	79.66
TS-R (P)	65.47	70.65	81.45	83.70	71.60
TS-R (R)	60.83	75.21	88.91	89.77	89.77
RS (No)	3571	3339	2096	**1404**	3086

Table 6: Diagnostic data sets results for selected models: 'DCS' - Document Context Sentences, 'TS-O' - Template Sentences with original entities, 'TS-R' - Template Sentences with replaced entities, 'RS' - Random Sentences. F1=multilabel F1-score, P=Precision, R=Recall, No=number of returned entities (lower is better). In the RS data set there are 2000 strings pretending to be sentences.

4 Discussion

On the basis of our research, we can draw a number of conclusions that are not often addressed to in publications about new neural models, their achievements and architecture. The scope of any assessment of new methods and models should be broadened to the understanding of their mistakes and the reasons why these models perform well or poorly in concrete examples, contexts and word meanings. These issues are particularly important in text data sets, in which semantic meaning and linguistic syntax are very complex.

In our effort to define linguistic categories for problematic Named Entities and their statistics in the CoNLL 2003 test set, we were able to draw a few additional conclusions regarding data annotation and augmentation processes. Moreover, our categories are similar to the taxonomy defined in publication about errors analysis for Uyghur Named Tagger (Abudukelimu et al., 2018).

4.1 The annotation process

The annotation process is a very tedious and exhaustive task for a person involved. Errors in data sets are expected but what must be checked is their impact on generalizing a model, e.g. one can create entities in places where they do not occur and check the model's stability. There are some useful applications for detecting annotation errors (Ratner et al., 2017), (Graliński et al., 2019) and (Wisniewski, 2018) but they are not used very often. Obviously, an appropriate and exhaustive documentation for the data set creation and annotation process is crucial. All annotated entity types should be described in details and examples of border cases should be given. In our analysis of the CoNLL 2003 data set we did not find any documentation. We have made our own assumptions and tried to guess why some classes are annotated in a given way. However, the work was hard and required many discussions and extended reviews of literature.

Secondly, there is a need for extended data sets with a broadened annotation process, similar to that of our diagnostic sets. E.g. linguists can extend their work not only just to the labelling of items (sentences), but also to indicating the scope of context that is necessary to recognise an entity, and to extending annotations for difficult cases or adding sub-types of entities.

Our work on diagnostic data sets is an attempt to extend an annotation process by focusing only on specific use cases which are less represented in the original data set.

4.2 Extended context

A new model training process itself should consist of more augmentation of the data set. Currently, there is some work being done on this topic, e.g. a semi-supervised context change with cutting the neighbourhood around NEs using a sliding window (Clark et al., 2018). Other techniques could be a random change of the first letter (or whole words) of NEs so that the model would not be so vulnerable to capitalized letters in names or small changes in sentences (e.g. adding or removing a dot at the end of a sentence).

Furthermore, a sentence itself is not always sufficient to recognise a class of a NE. In these cases, in both training and test data sets, there should be more samples where there are indications of co-references that are important to recognise particular NEs. Then, the input of a model should comprise a sentence and embedded features (or any representation) of co-references or their contexts. E.g. *"Little was banned. Peter Little took part in the last match with Welsh team."* - in the first sentence, we are are not sure if it is a NE. Then *"Peter Little"* indicates the proper NE type. An example of a model and data processing pipeline (i.e. memory of embeddings) that takes into consideration the same names in different sentences is to be found in (Akbik et al., 2019) and (Zhang et al., 2018).

Another important improvement is adding information about document layout or the structure of a text, e.g. a table, its rows and columns, and headings. In CoNLL 2003, there are many sports news, stock exchange reports or timetables where the structure of a text helps to understand its context, and thus to better recognise its NEs. Such a solution for another domaininvoice information extractionis elaborated on by (Katti et al., 2018) or (Liu et al., 2019). The solutions mentioned here combine character information with document image information in one architecture of a neural network.

The CoNLL 2003 test set is certainly too small to test the generalisation and stability of a model. Faced with this issue, we must find new techniques to prevent over-fitting. For instance, we could check a model's resistance to examples prepared in our diagnostics data sets, e.g. after changing a NE in a template sentence, the model should find the entity in the same place. We could also prepare small modifications to our original sentences, e.g. add or remove a dot at the end of an example and compare results (similarly to adversarial methods).

5 Concluding remarks

Mistakes are not all created equal. A comparison of models based on scores like F1 is rather simplistic. In this paper we defined 4 major and 11 minor linguistic categories of errors for NER problems. For the CoNLL 2003 data set and five important ML models (Stanford, CMU, ELMO, FLAIR, BERT-base) we re-annotated all errors with respect to the newly proposed ontology.

The presented analysis helps better understand a source of problems in recent models and also to better understand why some models are more reliable on one data set but less not on another.

Acknowledgements

TSt, AWr, AW, DZi would like to thank for the financial support that their project conducted by Applica.ai has received from the European Regional Development Fund (POIR.01.01.01-00- 0144/17-00). PBi was financially supported by European Regional Development Fund POIR.01.01.01-00-0328/17.

Our research was also partially supported as a part of the RENOIR Project by the European Unions Horizon 2020 Research and Innovation Programme under the Marie Skodowska-Curie grant agreement No 691152 and by the Ministry of Science and Higher Education (Poland), grant Nos. W34/H2020/2016.

References

Halidanmu Abudukelimu, Abudoukelimu Abulizi, Boliang Zhang, Xiaoman Pan, Di Lu, Heng Ji, and Yang Liu. 2018. Error analysis of Uyghur name tagging: Language-specific techniques and remaining challenges. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics NAACL*. Association for Computational Linguistics.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649. Association for Computational Linguistics.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. Mach. Learn. Res.*, 6:1817–1853.

Alexei Baevski, Sergey Edunov, Yinhan Liu, Luke Zettlemoyer, and Michael Auli. 2019. Cloze-driven pretraining of self-attention networks. *CoRR*, abs/1903.07785.

google bert. 2019. google-research/bert repository (issue 223).

Adrian Braşoveanu, Giuseppe Rizzo, Philipp Kuntschik, Albert Weichselbraun, and Lyndon J.B. Nixon. 2018. Framing named entity linking error types. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa.

2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Flair. 2018. Flair repository (issue 206 and 390).

Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named entity recognition through classifier combination. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*.

Filip Graliński, Anna Wróblewska, Tomasz Stanisławek, Kamil Grabowski, and Tomasz Górecki. 2019. GEval: Tool for debugging NLP datasets and models. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 254–262, Florence, Italy. Association for Computational Linguistics.

Masaaki Ichihara, Kanako Komiya, Tomoya Iwakura, and Maiko Yamazaki. 2015. Error analysis of named entity recognition in bccwj.

Anoop R. Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. 2018. Chargrid: Towards understanding 2d documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4459–4469. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2018. A survey on deep learning for named entity recognition. *CoRR*, abs/1812.09449.

Xiaojing Liu, Feiyu Gao, Qiong Zhang, and Huasha Zhao. 2019. Graph convolution for multimodal information extraction from visually rich documents. *CoRR*, abs/1903.11279.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074. Association for Computational Linguistics.

Christopher D. Manning. 2011. Part-of-speech tagging from 97% to 100%: Is it time for some linguistics? In *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text Processing - Volume Part I*, CICLing'11, pages 171–189, Berlin, Heidelberg. Springer-Verlag.

MUC. 1995. Muc-6 challenges and data sets.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 78–86. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason Alan Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. *CoRR*, abs/1711.10160.

Georges Rey. 2018. The analytic/synthetic distinction. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, fall 2018 edition. Metaphysics Research Lab, Stanford University.

Beth M. Sundheim. 1995. Overview of results of the muc-6 evaluation. In *Proceedings of the 6th Conference on Message Understanding*, MUC6 '95, pages 13–31, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*.

Guillaume Wisniewski. 2018. Errator: a tool to help detect annotation errors in the universal dependencies project. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158. Association for Computational Linguistics.

Boliang Zhang, Spencer Whitehead, Lifu Huang, and Heng Ji. 2018. Global attention for name tagging. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 86–96.

Appendix B: Kleister: Key information extraction datasets involving long documents with complex layouts

Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts

Tomasz Stanisławek[1,2], Filip Graliński[1,3(✉)], Anna Wróblewska[2], Dawid Lipiński[1], Agnieszka Kaliska[1,3], Paulina Rosalska[1,4], Bartosz Topolski[1], and Przemysław Biecek[2,5]

[1] Applica.ai, 15 Zajęcza, Warsaw 00351, Poland
{tomasz.stanislawek,dawid.lipinski,paulina.rosalska, bartosz.topolski}@applica.ai

[2] Warsaw University of Technology, Koszykowa 75, Warsaw, Poland
anna.wroblewska@pw.edu.pl

[3] Adam Mickiewicz University, 1 Wieniawskiego, Poznan 61712, Poland
{filip.gralinski,agnieszka.kaliska}@amu.edu.pl

[4] Nicolaus Copernicus University, 11 Gagarina, Torun 87100, Poland

[5] Samsung R&D Institute Poland, Plac Europejski 1, Warsaw, Poland
przemyslaw.biecek@samsung.com

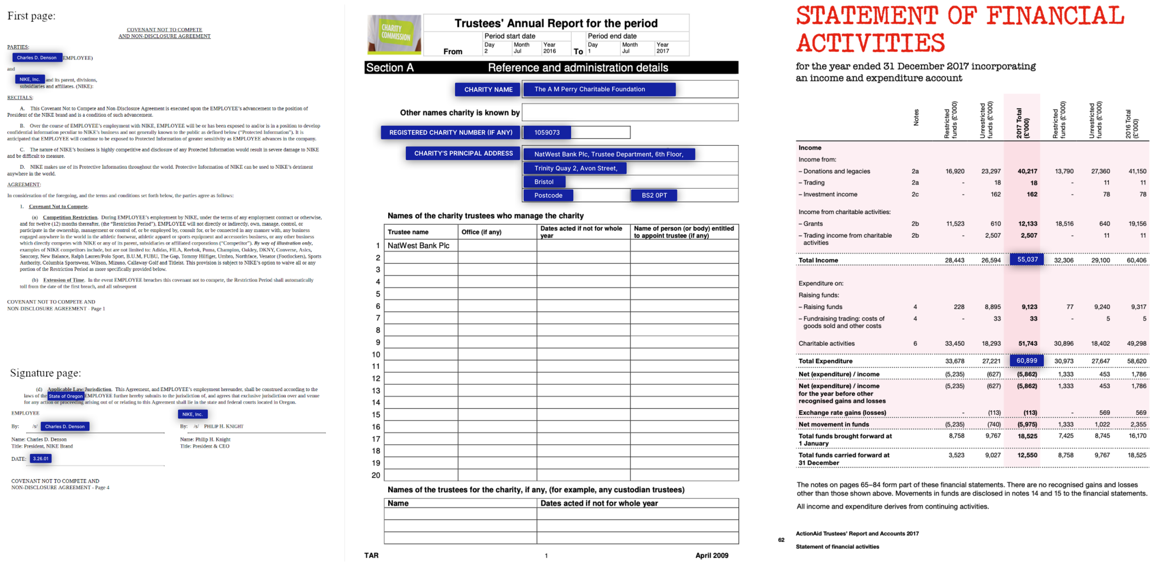
Abstract. The relevance of the Key Information Extraction (KIE) task is increasingly important in natural language processing problems. But there are still only a few well-defined problems that serve as benchmarks for solutions in this area. To bridge this gap, we introduce two new datasets (*Kleister NDA* and *Kleister Charity*). They involve a mix of scanned and born-digital long formal English-language documents. In these datasets, an NLP system is expected to find or infer various types of entities by employing both textual and structural layout features. The Kleister Charity dataset consists of 2,788 annual financial reports of charity organizations, with 61,643 unique pages and 21,612 entities to extract. The Kleister NDA dataset has 540 Non-disclosure Agreements, with 3,229 unique pages and 2,160 entities to extract. We provide several state-of-the-art baseline systems from the KIE domain (Flair, BERT, RoBERTa, LayoutLM, LAMBERT), which show that our datasets pose a strong challenge to existing models. The best model achieved an 81.77% and an 83.57% F1-score on respectively the Kleister NDA and the Kleister Charity datasets. We share the datasets to encourage progress on more in-depth and complex information extraction tasks.

Keyword: Key information extraction, visually rich documents, named entity recognition

1 Introduction

The task of Key Information Extraction (KIE) from Visually Rich Documents (VRD) has proved increasingly interesting in the business market with the recent

J. Lladós et al. (Eds.): ICDAR 2021, LNCS 12821, pp. 564–579, 2021.
https://doi.org/10.1007/978-3-030-86549-8_36



First page:

COVENANT NOT TO COMPETE
AND NON-DISCLOSURE AGREEMENT

PARTIES:

Charles D. Denson EMPLOYEE)

and

NIKE, Inc. and its parent, divisions, subsidiaries and affiliates. (NIKE):

RECITALS:

A. This Covenant Not to Compete and Non-Disclosure Agreement is executed upon the EMPLOYEE's advancement to the position of President of the NIKE brand and is a condition of such advancement.

B. Over the course of EMPLOYEE's employment with NIKE, EMPLOYEE will be or has been exposed to and/or is in a position to develop confidential information peculiar to NIKE's business and not generally known to the public as defined below ("Protected Information"). It is anticipated that EMPLOYEE will continue to be exposed to Protected Information of greater sensitivity as EMPLOYEE advances in the company.

C. The nature of NIKE's business is highly competitive and disclosure of any Protected Information would result in severe damage to NIKE and be difficult to measure.

D. NIKE makes use of its Protective Information throughout the world. Protective Information of NIKE can be used to NIKE's detriment anywhere in the world.

AGREEMENT:

In consideration of the foregoing, and the terms and conditions set forth below, the parties agree as follows:

1. **Covenant Not to Compete.**

(a) **Competition Restriction.** During EMPLOYEE's employment by NIKE, under the terms of any employment contract or otherwise, and for twelve (12) months thereafter, (the "Restriction Period"), EMPLOYEE will not directly or indirectly, own, manage, control, or participate in the ownership, management or control of, or be employed by, consult for, or be connected in any manner with, any business engaged anywhere in the world in the athletic footwear, athletic apparel or sports equipment and accessories business, or any other business which directly competes with NIKE or any of its parent, subsidiaries or affiliated corporations ("Competitor"). ***By way of illustration only,*** examples of NIKE competitors include, but are not limited to: Adidas, FILA, Reebok, Puma, Champion, Oakley, DKNY, Converse, Asics, Saucony, New Balance, Ralph Lauren/Polo Sport, B.U.M, FUBU, The Gap, Tommy Hilfiger, Umbro, Northface, Venator (Footlockers), Sports Authority, Columbia Sportswear, Wilson, Mizuno, Callaway Golf and Titleist. This provision is subject to NIKE's option to waive all or any portion of the Restriction Period as more specifically provided below.

(b) **Extension of Time.** In the event EMPLOYEE breaches this covenant not to compete, the Restriction Period shall automatically toll from the date of the first breach, and all subsequent

COVENANT NOT TO COMPETE AND
NON-DISCLOSURE AGREEMENT - Page 1

Signature page:

(d) **Applicable Law/Jurisdiction.** This Agreement, and EMPLOYEE's employment hereunder, shall be construed according to the laws of the State of Oregon EMPLOYEE further hereby submits to the jurisdiction of, and agrees that exclusive jurisdiction over and venue for any action or proceeding arising out of or relating to this Agreement shall lie in the state and federal courts located in Oregon.

EMPLOYEE	NIKE, Inc.
By: /s/ Charles D. Denson	By: /s/ PHILIP H. KNIGHT
Name: Charles D. Denson Title: President, NIKE Brand	Name: Philip H. Knight Title: President & CEO
DATE: 3.26.01	

COVENANT NOT TO COMPETE AND
NON-DISCLOSURE AGREEMENT - Page 4

CHARITY COMMISSION

Trustees' Annual Report for the period

	Period start date				Period end date		
From	Day 2	Month Jul	Year 2016	To	Day 1	Month Jul	Year 2017

Section A Reference and administration details

CHARITY NAME The A M Perry Charitable Foundation

Other names charity is known by

REGISTERED CHARITY NUMBER (IF ANY) 1059073

CHARITY'S PRINCIPAL ADDRESS NatWest Bank Plc, Trustee Department, 6th Floor,
Trinity Quay 2, Avon Street,
Bristol
Postcode BS2 0PT

Names of the charity trustees who manage the charity

	Trustee name	Office (if any)	Dates acted if not for whole year	Name of person (or body) entitled to appoint trustee (if any)
1	NatWest Bank Plc			
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Names of the trustees for the charity, if any, (for example, any custodian trustees)

Name	Dates acted if not for whole year

TAR 1 April 2009

STATEMENT OF FINANCIAL ACTIVITIES

for the year ended 31 December 2017 incorporating
an income and expenditure account

	Notes	Restricted funds (£'000)	Unrestricted funds (£'000)	2017 Total (£'000)	Restricted funds (£'000)	Unrestricted funds (£'000)	2016 Total (£'000)
Income							
Income from:							
– Donations and legacies	2a	16,920	23,297	**40,217**	13,790	27,360	41,150
– Trading	2a	-	18	**18**	-	11	11
– Investment income	2c	-	162	**162**	-	78	78
Income from charitable activities:							
– Grants	2b	11,523	610	**12,133**	18,516	640	19,156
– Trading income from charitable activities	2b	-	2,507	**2,507**	-	11	11
Total Income		28,443	26,594	55,037	32,306	29,100	60,406
Expenditure on:							
Raising funds:							
– Raising funds	4	228	8,895	**9,123**	77	9,240	9,317
– Fundraising trading: costs of goods sold and other costs	4	-	33	**33**	-	5	5
Charitable activities	6	33,450	18,293	**51,743**	30,896	18,402	49,298
Total Expenditure		33,678	27,221	60,899	30,973	27,647	58,620
Net (expenditure) / income		(5,235)	(627)	**(5,862)**	1,333	453	1,786
Net (expenditure) / income for the year before other recognised gains and losses		(5,235)	(627)	**(5,862)**	1,333	453	1,786
Exchange rate gains (losses)		-	(113)	**(113)**	-	569	569
Net movement in funds		(5,235)	(740)	**(5,975)**	1,333	1,022	2,355
Total funds brought forward at 1 January		8,758	9,767	**18,525**	7,425	8,745	16,170
Total funds carried forward at 31 December		3,523	9,027	**12,550**	8,758	9,767	18,525

The notes on pages 65–84 form part of these financial statements. There are no recognised gains and losses other than those shown above. Movements in funds are disclosed in notes 14 and 15 to the financial statements.

All income and expenditure derives from continuing activities.

62 **ActionAid Trustees' Report and Accounts 2017**
Statement of financial activities

Fig. 1. Examples of a real business applications and data for *Kleister* datasets. (Note: The key entities are in blue.) (Color figure online)

rise of solutions related to Robotic Process Automation (RPA). From a business user's point of view, systems that, fully automatically, gather information about individuals, their roles, significant dates, addresses and amounts, would be beneficial, whether the information is from invoices or receipts, from company reports or contracts [9,12,13,16,18,21,22]. There is a disparity between what can be delivered with the KIE domain systems on publicly available datasets and what is required by real-world business use. This disparity is still large and makes a robust evaluation difficult. Recently, researchers have started to fill the gap by creating datasets in the KIE domain such as scanned receipts: *SROIE*[1] [18], form understanding [11], NIST Structured Forms Reference Set of Binary Images (*SFRS*)[2] or Visual Question Answering dataset *DocVQA* [15].

This paper describes two new English-language datasets for the Key Information Extraction tasks from a diverse set of texts, long scanned and born-digital documents with complex layouts, that address real-life business problems (Fig. 1). The datasets represent various problems arising from the specificity of business documents and associated business conditions, e.g. complex layouts, specific business logic, OCR quality, long documents with multiple pages, noisy training datasets, and normalization. Moreover, we evaluate several systems from the KIE domain on our datasets and analyze KIE tasks' challenges in the business domain. We believe that our datasets will prove a good benchmark for more complex Information Extraction systems.

[1] https://rrc.cvc.uab.es/?ch=13&com=evaluation&task=3.

[2] https://www.nist.gov/srd/nist-special-database-2.

The main contributions of this study are:

1. *Kleister* – two novel datasets of long documents with complex layouts: 3,328 documents containing 64,872 pages with 23,772 entities to extract (see Sect. 3);
2. our method of collecting datasets using a semi-supervised methodology, which reduces the amount of manual work in gathering data; this method has the potential to be reused for similar tasks (see Sect. 3.1 and 3.2);
3. evaluation over several state-of-the-art Named Entity Recognition (NER) architectures (Flair, BERT, RoBERTa, LayoutLM, LAMBERT) employing our *Pipeline* method (see Sect. 4.1 and 5);
4. detailed analysis of the data and baseline results related to the Key Information Extraction task carried out by human experts (see Sect. 3.3 and 5).

The data, except for the test-set gold standard, are available at https://github.com/applicaai/kleister-nda.git and https://github.com/applicaai/kleister-charity.git. A shared-task platform where submissions can be evaluated, also for the test set, is available at https://gonito.applica.ai.

2 Related Work

Our main reason for preparing a new dataset was to develop a strategy to deal with challenges faced by businesses, which means overcoming such difficulties as complex layout, specific business logic (the way that content is formulated, e.g. tables, lists, titles), OCR quality, document-level extraction and normalization.

2.1 KIE from Visually Rich Documents (publicly Available)

A list of KIE-oriented challenges is available at the International Conference on Document Analysis and Recognition ICDAR 2019[3] (cf. Table 1). There is a dataset called SROIE[4] with information extraction from a set of scanned receipts. The authors prepared 1,000 whole scanned receipt images with annotated entities: company name, date, address, and total amount paid (a similar dataset was also created [18]). Form Understanding in Noisy Scanned Documents is another interesting dataset from ICDAR 2019 (*FUNSD*) [11]. FUNSD aims at extracting and structuring the textual content of forms. However, the authors focus mainly on understanding tables and a limited range of document layouts, rather than on extracting particular entities from the data. The point is, therefore, to indicate a table but not to extract the information it contains.

2.2 KIE from Visually Rich Documents (publicly Unavailable)

There are also datasets for the Key Information Extraction task based on invoices [9,12,16,17]. Documents of this kind contain entities like ‘Invoice date,’

[3] http://icdar2019.org/competitions-2/.

[4] https://rrc.cvc.uab.es/?ch=13.

‘Invoice number,’ ‘Net amount’ and ‘Vendor Name’, extracted using a combination of NLP and Computer Vision techniques. The reason for such a complicated multi-domain process is that spatial information is essential for properly understanding these kinds of documents. However, since they are usually short, the same information is relatively rarely repeated, and therefore there is no need for understanding the more extended context of the document. Nevertheless, those kinds of datasets are the most similar to our use case.

2.3 Information Extraction from One-Dimensional Documents

The *WikiReading* dataset [8] (and its variant *WikiReading Recycled* [6]) is a large-scale natural language understanding task. Here, the main goal is to predict textual values from the structured knowledge base, Wikidata, by reading the text of the corresponding Wikipedia articles. Some entities can be extracted from the given text directly, but some have to be inferred. Thus, as in our assumptions, the task contains a rich variety of challenging extraction sub-tasks and it is also well-suited for end-to-end models that must cope with longer documents.

Key Information Extraction is different from the Named Entity Recognition task (the *CoNLL 2003* NER challenge [20] being a well-known example). This is because: (1) retrieving spans is not required in KIE; (2) a system is expected to extract specific, actionable data points rather than general types of entities (such as people, organization, locations and “others” for CoNLL 2003).

Table 1. Summary of the existing English datasets and the Kleister sets. (*) For detailed description see Sect. 3.3.

Dataset name	CoNLL 2003	WikiReading	FUNSD	SROIE	Kleister NDA	Kleister charity
Source	Reuters news	Wikipedia	Forms	Receipts	EDGAR	UK charity Com.
Documents	1,393	4.7M	199	973	540	2,778
Pages	–	–	199	973	3,229	61,643
Entities	35,089	18M	9,743	3,892	2,160	21,612
Train docs	946	16.03M	149	626	254	1,729
Dev docs	216	1.89M	–	–	83	440
Test docs	231	0.95M	50	347	203	609
Input/Output on token level(*)	✓	✗	✓	✓	✗	✗
Long document(*)	✗	✓	✗	✗	✓	✓
Complex layout(*)	✗	✗	✓	✓	✓	✓
OCR(*)	✗	✗	✓	✓	✗	✓

3 Kleister: New Datasets

We collected datasets of long formal born-digital documents, namely US non-disclosure agreements (Kleister NDA) and a mixture of born-digital and (mostly) scanned annual financial reports of charitable foundations from the UK (Kleister Charity). These two datasets have been gathered in different ways due to their repository structures. Also, they were published on the Internet for different reasons. The crucial difference between them is that the NDA dataset was born-digital, but that the Charity dataset needed to be OCRed. Kleister datasets have a multi-modal input (PDF files and text versions of the documents) and a list of entities to be found.

3.1 NDA Dataset

The NDA Dataset contains Non-disclosure Agreements, also known as Confidentiality Agreements. They are legally binding contracts between two or more parties, where the parties agree not to disclose information covered by the agreement. The NDAs can take on various forms (e.g. contract attachments, emails), but they usually have a similar structure.

Data Collection Method. The NDAs were collected from the Electronic Data Gathering, Analysis and Retrieval system (EDGAR[5]) via Google's search engine. The original files were in an HTML format, but they were transformed into PDF files to keep processing simple and similar to that of other public datasets. Transformation was made using the `puppeteer` library.[6] Then, a list of entities was established (see Table 1).

Annotation Procedure. We annotated the whole dataset in two ways. Its first part, made up of 315 documents, was annotated by three annotators, except that only contexts with some similarity, pre-selected using methods based on semantic similarity (cf. [3]), were taken into account; this was to make the annotation faster and less-labor intensive. The second part, with 195 documents, was annotated entirely by hand. When preparing the dataset, we wanted to determine whether semantic similarity methods could be applied to limit the time it would take to perform annotation procedures; this solution was about 50% quicker than fully manual annotation. The annotations on all documents were then checked by a super-annotator, which ensured the annotation's excellent quality Cohen's κ (=0.971)[7]. Next, all entities were normalized according to the standards adopted by us, e.g. the `effective date` was standardized according to ISO 8601 i.e. YYYY-MM-DD[8].

Dataset Split. The Kleister NDA dataset contains a relatively small document count, so we decided to add more examples into the test split (about 38%) so as to be more accurate during the evaluation stage (see Table 1 for exact numbers).

[5] https://www.sec.gov/edgar.shtml.

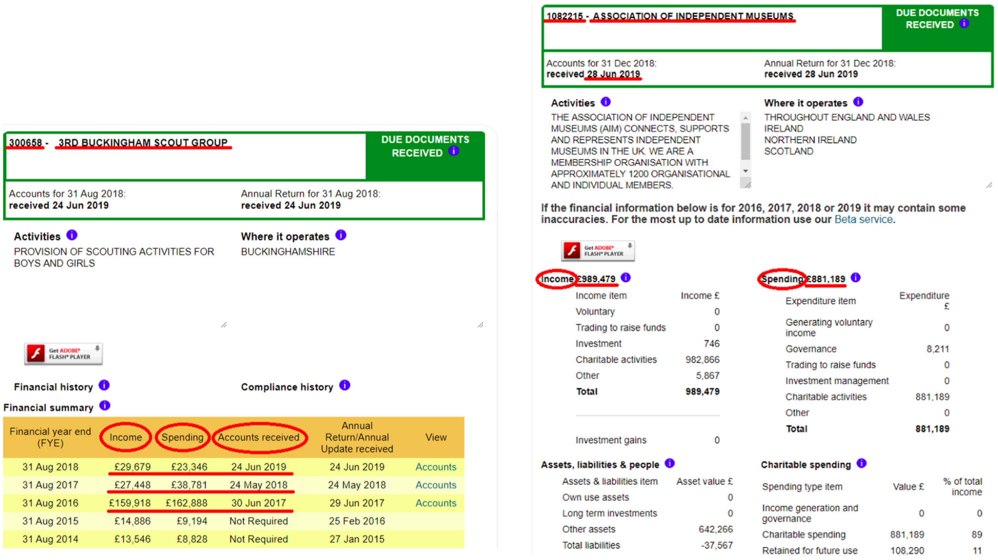
[6] https://github.com/puppeteer/puppeteer.

[7] https://en.wikipedia.org/wiki/Cohen%27s_kappa.

[8] The normalization standards are described in the public repository with datasets.

3.2 Charity Dataset

The Charity dataset consists of annual financial reports that all charities registered in England and Wales must submit to the Charity Commission. The Commission subsequently makes them publicly available on its website.[9] There are no strict rules about the format of these charity reports. Some are richly illustrated with photos and charts and financial information constitutes only a small part of the entire report. In contrast, others are a few pages long and only necessary data on revenues and expenses in a given calendar year are given.



300658 - 3RD BUCKINGHAM SCOUT GROUP

DUE DOCUMENTS RECEIVED

Accounts for 31 Aug 2018:
received 24 Jun 2019

Annual Return for 31 Aug 2018:
received 24 Jun 2019

Activities
PROVISION OF SCOUTING ACTIVITIES FOR BOYS AND GIRLS

Where it operates
BUCKINGHAMSHIRE

Financial history

Compliance history

Financial summary

Financial year end (FYE)	Income	Spending	Accounts received	Annual Return/Annual Update received	View
31 Aug 2018	£29,679	£23,346	24 Jun 2019	24 Jun 2019	Accounts
31 Aug 2017	£27,448	£38,781	24 May 2018	24 May 2018	Accounts
31 Aug 2016	£159,918	£162,888	30 Jun 2017	29 Jun 2017	Accounts
31 Aug 2015	£14,886	£9,194	Not Required	25 Feb 2016	
31 Aug 2014	£13,546	£8,828	Not Required	27 Jan 2015	

1082215 - ASSOCIATION OF INDEPENDENT MUSEUMS

DUE DOCUMENTS RECEIVED

Accounts for 31 Dec 2018:
received 28 Jun 2019

Annual Return for 31 Dec 2018:
received 28 Jun 2019

Activities
THE ASSOCIATION OF INDEPENDENT MUSEUMS (AIM) CONNECTS, SUPPORTS AND REPRESENTS INDEPENDENT MUSEUMS IN THE UK. WE ARE A MEMBERSHIP ORGANISATION WITH APPROXIMATELY 1200 ORGANISATIONAL AND INDIVIDUAL MEMBERS.

Where it operates
THROUGHOUT ENGLAND AND WALES
IRELAND
NORTHERN IRELAND
SCOTLAND

If the financial information below is for 2016, 2017, 2018 or 2019 it may contain some inaccuracies. For the most up to date information use our Beta service.

Income £989,479

Income item	Income £
Voluntary	0
Trading to raise funds	0
Investment	746
Charitable activities	982,866
Other	5,867
Total	**989,479**
Investment gains	0

Spending £881,189

Expenditure item	Expenditure £
Generating voluntary income	0
Governance	8,211
Trading to raise funds	0
Investment management	0
Charitable activities	881,189
Other	0
Total	**881,189**

Assets, liabilities & people

Assets & liabilities item	Asset value £
Own use assets	0
Long term investments	0
Other assets	642,266
Total liabilities	-37,567

Charitable spending

Spending type item	Value £	% of total income
Income generation and governance	0	0
Charitable spending	881,189	89
Retained for future use	108,290	11

Fig. 2. Organization's page on the Charity Commission's website (left: organization whose annual income is between 25k and 500k GBP, right: over 500k). Note: Entities are underlined in red and names of entities are circled.

Data Collection Method. The Charity Commission website has a database of all the charity organizations registered in England and Wales. Each of these organizations has a separate sub-page on the Commission's website, and it is easy to find the most important information about them there (see Fig. 2). This information only partly overlaps with information in the reports. Some entities such as, say, a list of trustees might not be in the reports. Thus, we decided to extract only those entities which also appear in the form of a brief description on the website.

In the beginning, we downloaded 3,414 reports (as PDF files).[10] During document analysis, it emerged that several reports were written in Welsh. As we are interested only in English, all documents in other languages were identified and

[9] https://apps.charitycommission.gov.uk/showcharity/registerofcharities/RegisterHomePage.aspx.

[10] Organizations with an income below 25,000 GBP a year are required to submit a condoned financial report instead.

removed from the collection. Additionally, documents that contained reports for more than one organization or whose OCR quality was low were deleted. This left us with 2,778 documents.

Annotation Procedure. There was no need to manually annotate all documents because information about the reporting organizations could be obtained directly from the Charity Commission. Initially, only a random sample of 100 documents were manually checked. Some proved low quality: `charity name` (5% of errors and 13% of minor differences), and `charity address` (9% of errors and 63% of minor differences). Minor errors are caused by data presentation differences on the page and in the document. For example, the charity's name on the website and in the document could be written with the term *Limited* (shortened to *LTD*) or without it. These minor differences were corrected manually or automatically. In the next step, 366 documents were analyzed manually. Some parts of the charity's address were also problematic. For instance, counties, districts, towns and cities were specified on the website, but not in the documents, or *vice versa*. We split the address data into three separate entities that we considered the most essential: postal code, postal town name and street or road name. The postal code was the critical element of the address, based on the city name and street name[11]. The whole process allowed us to accurately identify entities (see Table 1) and to obtain a good-quality dataset with annotations corresponding to the gold standard.

Dataset Split. In the Kleister Charity dataset, we have multiple documents from the same charity organization but from different years. Therefore, we decided to split documents based on charity organization into the train/dev/test sets with, respectively, a 65/15/20 dataset ratio (see Table 1 for exact numbers). The documents from the dev/test split were manually annotated (by two annotators) to ensure high-quality evaluation. Additionally, 100 random documents from the test set were annotated twice to calculate the relevant Cohen's κ coefficient (we achieved excellent quality $\kappa = 0.9$).

3.3 Statistics and Analysis

The detailed statistics of the Kleister datasets are presented in Table 1 and Table 2. Our datasets covered a broad range of general types of entities; the `party` entity is special since it could be one of the following types: `ORGANIZATION` or `PERSON`. Additionally, some documents may not contain all entities mentioned in the text, for instance in Kleister NDA the `term` entity appears in 36% of documents. Likewise, some entities may have more than one gold value; for instance in Kleister NDA the `party` entity could have up to 7 gold values for a single document. `Report_date`, `jurisdiction` and `term` have the lowest number of unique values. This suggests that these entities should be simpler than others to extract.

[11] Postal codes in the UK were aggregated from www.streetlist.co.uk.

Table 2. Summary of the entities in the NDA and charity datasets. (*) Based on manual annotation of text spans.

Entities	General entity type	Total count	Unique values	(*) Avg. entity count/doc	(*) Avg. token count/entity	Example gold value
NDA dataset (540 documents)						
Party	ORG/PER	1,035	912	19.74	1.62	Ajinomoto Althea Inc.
Jurisdiction	LOCATION	531	37	1.05	1.21	New York
Effective_date	DATE	400	370	1.95	3.10	2005-07-03
Term	DURATION	194	22	1.03	2.77	P12M
Charity dataset (2 788 documents)						
Post_town	ADDRESS	2,692	501	1.12	1.06	BURY
Postcode	ADDRESS	2,717	1,511	1.12	1.99	BL9 ONP
Street_line	ADDRESS	2,414	1,353	1.12	2.52	42–47 MINORIES
Charity_name	ORG	2,778	1,600	13.80	3.67	Mad Theatre Company
Charity_number	NUMBER	2,763	1,514	2.47	1.00	1143209
Report_date	DATE	2,776	129	10.58	2.96	2016-09-30
Income	AMOUNT	2,741	2,726	1.95	1.01	109370.00
Spending	AMOUNT	2,731	2,712	2.03	1.01	90174.00

Manual Annotation of Text Spans. To give more detailed statistics we decided to annotate small numbers of documents on text span level. Four annotators annotated 60/55 documents for, respectively, the Kleister Charity and Kleister NDA. In Table 2, we observe that 5 out of 12 entities appear once in a single document. There are also three entities with more than ten counts on average (`party`, `charity_number` and `report_date`). Annotation on the text-span level

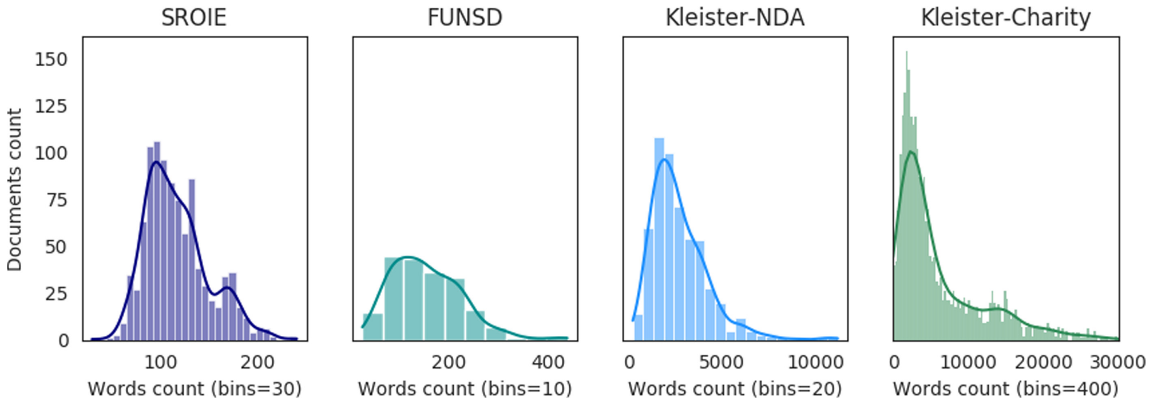


Fig. 3. Distribution of document lengths for kleister datasets compared to other similar datasets (note that the x-axes ranges are different).

could prove critical to checking the quality of the training dataset for methods based on a Named Entity Recognition model, something which an *autotagging* mechanism produces (see Sect. 4.1).

Comparison with Existing Resources. In Table 1, we gathered the most important information about open datasets (which are the most popular ones in the domain) and the Kleister datasets. In particular, we outlined the difference based on the following properties:

- **Input/Output on token level**: it is known which tokens an entity is made up from in the documents. Otherwise, one should: a) create a method for preparing a training dataset for sequence labeling models (subsequently in the publication, we use the term *autotagging* for this sub-task); b) infer or create a canonical form of the final output in order to deal with differences between the target entities provided in the annotations and their variants occurring in the documents (e.g. for `jurisdiction` we must transform a text-level span *NY* into a document-level gold value **New York**).
- **Long Document**: Fig. 3 presents differences in document lengths (calculated as a number of OCRed words) in the Kleister datasets compared to other similar datasets. Since entities could appear in documents multiple times in different contexts, we must be able to understand long documents as a whole. This leads, of course, to different architectural decisions [2,4]. For example, the `term` entity in the Kleister NDA dataset tells us about the contract duration. This information is generally found in the *Term* chapter, in the middle part of a document. However, sometimes we could also find a `term` entity at the end of the document, the task is to find out which of the values is correct.
- **Complex Layout**: this requires proper understanding of the complex layout (e.g. interpreting tables and forms as 2D structures), see Fig. 1.
- **OCR**: processing of scanned documents in such a way as to deal with possible OCR errors caused by handwriting, pages turned upside down or more general poor scan quality.

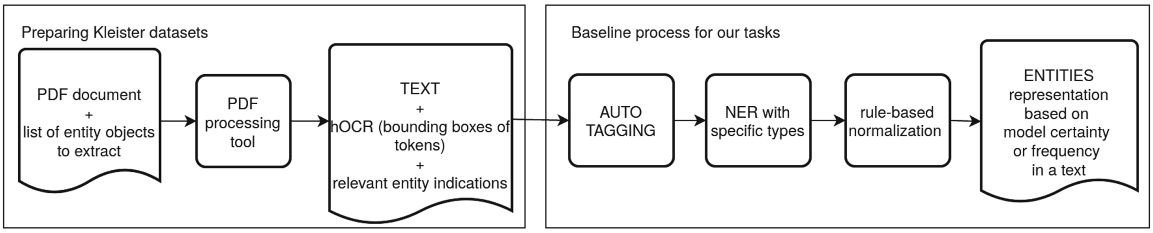

Preparing Kleister datasets
PDF document + list of entity objects to extract
PDF processing tool
TEXT + hOCR (bounding boxes of tokens) + relevant entity indications
Baseline process for our tasks
AUTO TAGGING
NER with specific types
rule-based normalization
ENTITIES representation based on model certainty or frequency in a text

Fig. 4. Our preparation process for Kleister datasets and training baselines. Initially, we gathered PDF documents and expected entities' values. Then, based on textual and layout features, we prepared our pipeline solutions. The pipeline process is illustrated in the second frame and consists of the following stages: autotagging; standard NER; text normalization; and final selection of the values of entities.

4 Experiments

Kleister datasets for the Key Information Extraction task are challenging and hardly any solutions in the current NLP world can solve them. In this experiment, we aim to produce strong baselines with the *Pipeline* approach (Sect. 4.1) to solve extraction problems. This method's core idea is to select specific parts of the text in a document that denote the objects we search for. The whole process is a chain of techniques with, crucially, a named entity recognition model: once indicated in a document (multiple occurrences are possible), entities are normalized, then all results are aggregated into the one value specified for a given entity type.

4.1 Document Processing Pipeline

Figure 4 presents the whole process, and all the stages are described below (a similar methodology was proposed [17]).

Autotagging. Since we have only document-level annotation in the Kleister datasets, we need to generate a training set for an NER model which takes text span annotation as the input. This stage involves extracting all the fragments that refer to the same or to different entities by using sets of regular expressions combined with a gold-standard value for each general entity type, e.g. date, organization and amount. In particular, when we try to detect a `report_date` entity, we must handle different date formats: 'November 29, 2019', '11/29/19' or '11-29-2019'. This step is performed only for the purpose of training (to get data for training a sequence labeler; it is not applied during the prediction). The quality of this step varies across entity types (see details in Table 3).

Named Entity Recognition. We trained a NER model on the autotagged dataset using one of the state-of-the-art (1) architectures working on plain text such as Flair [1], BERT-base [5], RoBERTa-base [14], or (2) models employing layout features (LayoutLM-base [23] and LAMBERT [7]). Then, at the evaluation stage, we use the NER model to detect all entity occurrences in the text.

Normalization. At this stage objects are normalized to the canonical form, which we have defined in the Kleister datasets. We use almost the same regular expression as during autotagging. For instance, all detected `report_date` occurrences are normalized. So 'November 29, 2019', '11/29/19' and '11-29-2019' are rendered in our standard '2019-11-29' form (ISO 8601).

Aggregation. The NER model might return more than one text span for a given entity, sometimes these are multiple occurrences of the same correct information. Sometimes these represent errors of the NER model. In any case, we need to produce a single output from multiple candidates detected by the NER model. We take a simple approach: all candidates are grouped by the extracted entities' normalized forms and for each group we sum up the scores and finally we return the values with the largest sums.

4.2 Experimental Setup

Due to the Kleister document's length, most currently available models limit input size and so are unable to process the documents in a single pass. Therefore, each document was split into 300-word chunks (for Flair) or 510 BPE tokens (for BERT/RoBERTa/LayoutLM/LAMBERT) with overlapping parts. The results from overlapping parts were aggregated by averaging all the scores obtained for each token in the overlap.

For the Flair-based pipeline, we used implementation from the Flair library [1] in version 0.6.1 with the following parameters: *learning rate* = 0.1, *batch size* = 32, *hidden size* = 256, *epoch* = 30/15 (resp. NDA and Charity), *patience* = 3, *anneal factor* = 0.5, and with a CRF layer on top. For pipeline based on BERT/RoBERTa/LayoutLM, we used the implementation from *transformers* [10] library in version 3.1.0 with the following parameters: *learning rate* = 2e−5, *batch size* = 8, *epoch* = 20, *patience* = 2. For pipeline based on LAMBERT model we used implementation shared by authors of the publication [7] and the same parameters as for the BERT/RoBERTa/LayoutLM models. All experiments were performed with the same settings.

Moreover, in our experiments, we tried different PDF processing tools for text extraction from PDF documents to check the importance of text quality for the final pipeline score:

- **Microsoft Azure Computer Vision API (Azure CV)**[12] – commercial OCR engine, version 3.0.0;
- **pdf2djvu/djvu2hocr**[13]– a free tool for object and text extraction from born-digital PDF files (this is not an OCR engine, hence it could be applied only to Kleister NDA), version 0.9.8;
- **Tesseract**[19] – this is the most popular free OCR engine currently available, we used version 4.1.1-rc1-7-gb36c.[14];
- **Amazon Textract**[15] – commercial OCR engine.

5 Results

Table 3 shows the results for the two Kleister datasets obtained with the Pipeline method for all tested models. The weakest model from our baselines is, in general, BERT, with a slight advantage in Kleister NDA over the Flair model and a large performance drop on Kleister Charity in comparison to others. The best model is LAMBERT, which improved the overall F_1-score with 0.77 and 2.04 for,

[12] https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-recognizing-text.

[13] http://jwilk.net/software/pdf2djvu, https://github.com/jwilk/ocrodjvu.

[14] run with `--oem 2 -l eng --dpi 300` flags (meaning both new and old OCR engines were used simultaneously, with language and pixel density set to English and 300dpi respectively).

[15] https://aws.amazon.com/textract/ (API in version from March 1, 2020 was used).

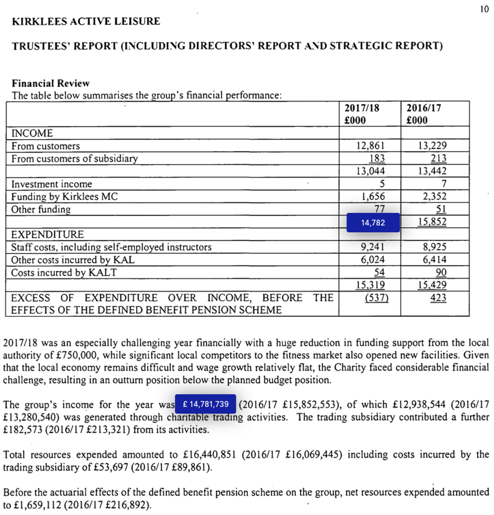
Table 3. The detailed results (average F_1-scores over 3 runs) of our baselines for Kleister challenges (test sets) for the best PDF processing tool. Autotagger F_1-scores were calculated based on results from our regexp mechanism and manual annotation on the text span level (see Sect. 3.3). Human performance is a percentage of annotators agreements for 100 random documents. We used the Base version of the BERT, RoBERTa, LayoutLM and LAMBERT models.

Kleister NDA dataset (pdf2djvu)							
Entity name	Flair	BERT	RoBERTa	LayoutLM	LAMBERT	Autotagger	Human
Effective_date	79.37	80.20	81.50	80.50	**85.27**	79.00	100%
Party	70.13	71.60	**80.83**	76.60	78.70	33.15	98%
Jurisdiction	93.87	95.00	92.87	94.23	**96.50**	54.10	100%
Term	**60.33**	45.73	52.27	47.63	55.03	74.10	95%
ALL	77.83	78.20	81.00	78.47	**81.77**	60.09	97.86%
Kleister Charity dataset (Azure CV)							
Post_town	83.07	77.03	77.70	76.57	**83.70**	66.04	98%
Postcode	89.57	87.10	88.40	88.53	**90.37**	87.60	100%
Street_line	69.10	62.23	72.03	70.92	**74.30**	75.02	96%
Charity_name	72.97	75.93	78.03	**79.63**	77.83	67.00	99%
Charity_number	96.60	**96.67**	95.37	96.13	95.80	98.60	98%
Income	70.67	67.30	69.73	70.40	**74.70**	69.00	97%
Report_date	95.93	96.60	96.77	96.40	**96.80**	89.00	100%
Spending	68.13	64.43	68.60	68.57	**74.20**	73.00	92%
ALL	81.17	78.33	81.50	81.53	**83.57**	78.16	97.45%

respectively, NDA and Charity. It is worth noting that for born-digital documents in Kleister NDA this difference is not substantial. This is due to the fact that only for `effective_date` entity does the LAMBERT model have a clear advantage (about 4 points gain of F_1-score) over other baseline models. For Kleister Charity LAMBERT achieves the biggest improvement over sequential models on `income` (+4.03) and `spending` (+5.60) which appears mostly in table-like structures.

The most challenging problems for all models are entities (`effective_date`, `party`, `term`, `post_town`, `postcode`, `street_line`, `charity_name`, `income`, `spending`) related to the properties described in Sect. 3.3.

Input/Output on Token Level (Autotagging). As we can observe in Table 3, our autotagging mechanism with information about entity achieves, on the text span level, a performance inferior to almost all our models on the document level. It shows that, despite the fact that the autotagging mechanism is prone to errors, we could train a good quality NER model. Our analysis shows that there are some specific issues related to a regular-expression-based mechanism, e.g. `party` in the Kleister NDA dataset has the lowest score because organization names often occur in the text as an acronym or as a shortened form; for instance for `party` entity text *Emerson Electric Co.* means the same as *Emerson.* This is not easy to capture with a general regexp rule.



10

KIRKLEES ACTIVE LEISURE

TRUSTEES' REPORT (INCLUDING DIRECTORS' REPORT AND STRATEGIC REPORT)

Financial Review
The table below summarises the group's financial performance:

	2017/18 £000	2016/17 £000
INCOME		
From customers	12,861	13,229
From customers of subsidiary	183	213
	13,044	13,442
Investment income	5	7
Funding by Kirklees MC	1,656	2,352
Other funding	77	51
	14,782	15,852
EXPENDITURE		
Staff costs, including self-employed instructors	9,241	8,925
Other costs incurred by KAL	6,024	6,414
Costs incurred by KALT	54	90
	15,319	15,429
EXCESS OF EXPENDITURE OVER INCOME, BEFORE THE EFFECTS OF THE DEFINED BENEFIT PENSION SCHEME	(537)	423

2017/18 was an especially challenging year financially with a huge reduction in funding support from the local authority of £750,000, while significant local competitors to the fitness market also opened new facilities. Given that the local economy remains difficult and wage growth relatively flat, the Charity faced considerable financial challenge, resulting in an outturn position below the planned budget position.

The group's income for the year was £ 14,781,739 (2016/17 £15,852,553), of which £12,938,544 (2016/17 £13,280,540) was generated through charitable trading activities. The trading subsidiary contributed a further £182,573 (2016/17 £213,321) from its activities.

Total resources expended amounted to £16,440,851 (2016/17 £16,069,445) including costs incurred by the trading subsidiary of £53,697 (2016/17 £89,861).

Before the actuarial effects of the defined benefit pension scheme on the group, net resources expended amounted to £1,659,112 (2016/17 £216,892).

Fig. 5. Normalization issues for an `income` entity (amount in the table should be multiplied by 1000).

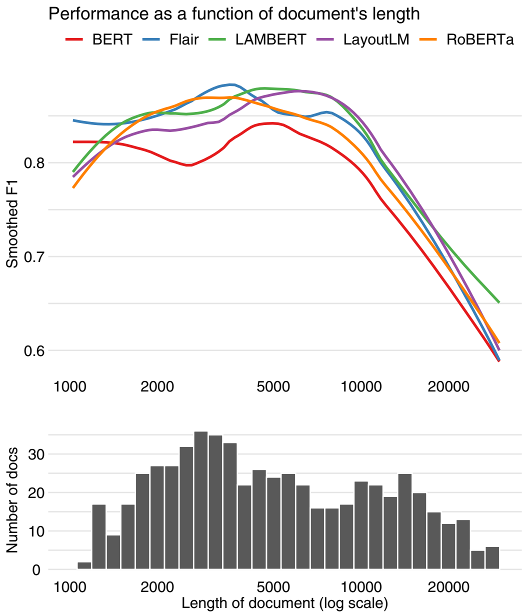


Fig. 6. Relationship between F_1-scores and document length in the Kleister Charity test set for the Azure CV OCR.

Input/Output on Token Level (normalization). We found that we could not achieve competitive results by using models based only on sequence labeling. For example, for the entities `income` and `spending` in the Kleister Charity dataset, we manually checked that in about 5% of examples we need to also infer the right scale (thousand, million, etc.) for each monetary value based on the document context (see Fig. 5).

Long Documents. It turns out that, for all models, worse results are observed for longer documents, see Fig. 6.

Complex Layout. The LAMBERT model has proved the best one, which proved the importance of using models employing not only textual (1D) but also layout (2D) features (see Table 3). Additionally, we also observe that the entities appearing in the sequential contexts achieve higher F_1-scores (`charity_number` and `report_date` entities in the Kleister Charity dataset).

OCR. We present the importance of using a PDF processing tool of good quality (see Table 4). With such a tool, we could gain several points in the F_1-score. There are two main conclusions: 1) Commercial OCR engines (Azure CV and Textract) are significantly better than Tesseract for scanned documents (Kleister Charity dataset). This is especially for true for 1D models not trained on Tesseract output (Flair, BERT, RoBERTa); 2) If we have the means to detect born-digital PDF documents, we should process them with a dedicated PDF tool (such as pdf2djvu) instead of using an OCR engine.

Table 4. F_1-scores for different PDF processing tools and models checked on Kleister challenges test sets over 3 runs with standard deviation. (*) pdf2djvu does not work on scans. We used the Base version of the BERT, RoBERTa, LayoutLM and LAMBERT models.

Kleister NDA dataset (born-digital PDF files)					
PDF tool	Flair	BERT	RoBERTa	LayoutLM	LAMBERT
Azure CV	$78.03_{\pm 0.12}$	$77.67_{\pm 0.18}$	$79.33_{\pm 0.68}$	$77.43_{\pm 0.29}$	$80.57_{\pm 0.25}$
pdf2djvu	$77.83_{\pm 0.26}$	$78.20_{\pm 0.17}$	$81.00_{\pm 0.05}$	$78.47_{\pm 0.76}$	$\mathbf{81.77_{\pm 0.09}}$
Tesseract	$76.57_{\pm 0.49}$	$76.60_{\pm 0.30}$	$77.81_{\pm 0.97}$	$77.70_{\pm 0.48}$	$81.03_{\pm 0.23}$
Textract	$77.37\ _{\pm 0.08}$	$74.83_{\pm 0.45}$	$79.49_{\pm 0.32}$	$77.40_{\pm 0.40}$	$77.37_{\pm 0.08}$
Kleister Charity dataset (mixture of born-digital and scanned PDF files) (*)					
Azure CV	$81.17_{\pm 0.12}$	$78.33_{\pm 0.08}$	$81.50_{\pm 0.23}$	$81.53_{\pm 0.23}$	$\mathbf{83.57_{\pm 0.29}}$
Tesseract	$72.87_{\pm 0.81}$	$71.37_{\pm 1.25}$	$76.23_{\pm 0.15}$	$77.53_{\pm 0.20}$	$81.50_{\pm 0.07}$
Textract	$78.03_{\pm 0.12}$	$73.30_{\pm 0.43}$	$80.08_{\pm 0.15}$	$80.23_{\pm 0.41}$	$82.97_{\pm 0.21}$

6 Conclusions

In this paper, we introduced two new datasets Kleister NDA and Kleister Charity for Key Information Extraction tasks. We set out in detail the process necessary for the preparation of these datasets. Our intention was to show that Kleister datasets will help the NLP community to investigate the effects of document lengths, complex layouts, and OCR quality problems on KIE performance.

We prepared baseline solutions based on text and layout data generated by different PDF processing tools from the datasets. The best model from our baselines achieves 81.77/83.57 F_1-score for, respectively, the Kleister NDA and Charity, which is much lower in comparison to datasets in a similar domain (e.g. 98.17 [7] for SROIE). This benchmark shows the weakness of the currently available state-of-the-art models for the Key Information Extraction task.

Acknowledgments.. The Smart Growth Operational Programme supported this research under project no. POIR.01.01.01-00-0605/19 (*Disruptive adoption of Neural Language Modelling for automation of text-intensive work*).

References

1. Akbik, A., Blythe, D., Vollgraf, R.: Contextual string embeddings for sequence labeling. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1638–1649. Association for Computational Linguistics, Santa Fe, New Mexico, USA (August 2018), https://www.aclweb.org/anthology/C18-1139
2. Beltagy, I., Peters, M.E., Cohan, A.: Longformer: The long-document transformer. ArXiv arXiv:2004.05150 (2020)

3. Borchmann, L., et al.: Contract discovery: Dataset and a few-shot semantic retrieval challenge with competitive baselines. In: Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16–20 November 2020, pp. 4254–4268. Association for Computational Linguistics (2020)
4. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q., Salakhutdinov, R.: Transformer-xl: Attentive language models beyond a fixed-length context. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (2019). https://www.aclweb.org/anthology/P19-1285
5. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. ArXiv arXiv:1810.04805 (2018)
6. Dwojak, T., Pietruszka, M., Borchmann, Ł., Chłędowski, J., Graliński, F.: From dataset recycling to multi-property extraction and beyond. In: Proceedings of the 24th Conference on Computational Natural Language Learning, pp. 641–651. Association for Computational Linguistics, Online (November 2020). https://doi.org/10.18653/v1/2020.conll-1.52, https://www.aclweb.org/anthology/2020.conll-1.52
7. Garncarek, Ł., et al.: LAMBERT: Layout-Aware (Language) Modeling using BERT for information extraction. ArXiv arXiv:2002.08087 (2020)
8. Hewlett, D., et al.: WikiReading: a novel large-scale language understanding task over Wikipedia. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1535–1545. Association for Computational Linguistics, Berlin, Germany (2016)
9. Holt, X., Chisholm, A.: Extracting structured data from invoices. In: Proceedings of the Australasian Language Technology Association Workshop 2018, pp. 53–59. Dunedin, New Zealand (December 2018). https://www.aclweb.org/anthology/U18-1006
10. Hugging Face: Transformers. https://github.com/huggingface/transformers (2020)
11. Jaume, G., Kemal Ekenel, H., Thiran, J.: FUNSD: A dataset for form understanding in noisy scanned documents. In: 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), vol. 2, pp. 1–6 (2019)
12. Katti, A.R., Reisswig, C., Guder, C., Brarda, S., Bickel, S., Höhne, J., Faddoul, J.B.: Chargrid: Towards Understanding 2D Documents. ArXiv arXiv:1809.08799 (2018)
13. Liu, X., Gao, F., Zhang, Q., Zhao, H.: Graph convolution for multimodal information extraction from visually rich documents. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (2019). http://dx.doi.org/10.18653/v1/N19-2005
14. Liu, Y., et al.: RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv arXiv:1907.11692 (2019)
15. Mathew, M., Karatzas, D., Jawahar, C.V.: DocVQA: A Dataset for VQA on Document Images. ArXiv arXiv:2007.00398 (2021)
16. Palm, R.B., Laws, F., Winther, O.: Attend, copy, parse end-to-end information extraction from documents. In: International Conference on Document Analysis and Recognition (ICDAR) (2019)
17. Palm, R.B., Winther, O., Laws, F.: Cloudscan - a configuration-free invoice analysis system using recurrent neural networks. In: 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) (2017). https://doi.org/10.1109/icdar.2017.74
18. Park, S., et al.: CORD: a consolidated receipt dataset for post-OCR parsing. In: Document Intelligence Workshop at Neural Information Processing Systems (2019)

19. Smith, R.: Tesseract Open Source OCR Engine (2020). https://github.com/tesseract-ocr/tesseract
20. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In: Proceedings of the Seventh Conference of the North American Chapter of the Association for Computational Linguistics (2003)
21. Wellmann, C., Stierle, M., Dunzer, S., Matzner, M.: A framework to evaluate the viability of robotic process automation for business process activities. In: Asatiani, A., et al. (eds.) BPM 2020. LNBIP, vol. 393, pp. 200–214. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58779-6_14
22. Wróblewska, A., Stanisławek, T., Prus-Zajączkowski, B., Garncarek, Ł.: Robotic process automation of unstructured data with machine learning. In: Position Papers of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, 9–12 September 2018, pp. 9–16 (2018). https://doi.org/10.15439/2018F373
23. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: LayoutLM: pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (2020). https://doi.org/10.1145/3394486.3403172

Appendix C: DUE: End-to-End Document Understanding Benchmark

DUE: End-to-End Document Understanding Benchmark

Łukasz Borchmann[*] **Michał Pietruszka**[*], **Tomasz Stanisławek**[*]
Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, Filip Graliński
Applica.ai
`firstname.surname@applica.ai`

Abstract

Understanding documents with rich layouts plays a vital role in digitization and hyper-automation but remains a challenging topic in the NLP research community. Additionally, the lack of a commonly accepted benchmark made it difficult to quantify progress in the domain. To empower research in this field, we introduce the Document Understanding Evaluation (DUE) benchmark consisting of both available and reformulated datasets to measure the end-to-end capabilities of systems in real-world scenarios. The benchmark includes Visual Question Answering, Key Information Extraction, and Machine Reading Comprehension tasks over various document domains and layouts featuring tables, graphs, lists, and infographics. In addition, the current study reports systematic baselines and analyzes challenges in currently available datasets using recent advances in layout-aware language modeling. We open both the benchmarks and reference implementations and make them available at `http://duebenchmark.com`.

1 Introduction

While mainstream Natural Language Processing focuses on plain text documents, the content one encounters when reading, e.g., scientific articles, company announcements, or even personal notes, is seldom plain and purely sequential. In particular, the document's visual and layout aspects that guide our reading process and carry non-textual information appear to be an essential aspect that requires comprehension. These layout aspects, as we understand them, are prevalent in tasks that can be much better solved when given not only sequence text on the input but pieces of multimodal information covering aspects such as text-positioning (i.e. location of words on the 2D plane), text-formatting (e.g., different font sizes, colors), and graphical elements (e.g., lines, bars, presence of figures) among others. Over the decades, systems dealing with document understanding developed an inherent aspect of multi-modality that nowadays revolves around the problems of integrating visual information with spatial relationships and text [33, 2, 47, 12].

In general, when document processing systems are considered, the term *understanding* is thought of specifically as the capacity to convert a document into meaningful information [9, 54, 15]. This fits into the rapidly growing market of hyperautomation-enabling technologies, estimated to reach nearly $600 billion in 2022, up 24% from 2020 [39]. Considering that unstructured data is orders of magnitude more abundant than structured data, the lack of necessary tools to analyze unstructured data and extract structured information can limit the performance of these intelligent services. The process of structuring data and content must be robust to various document domains and tasks.

Despite its importance for digital transformation, the problem of measuring how well available models obtain information from a wide range of tasks and document types and how suitable they are

[*] Equal contribution

Submitted to the 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks. Do not distribute.

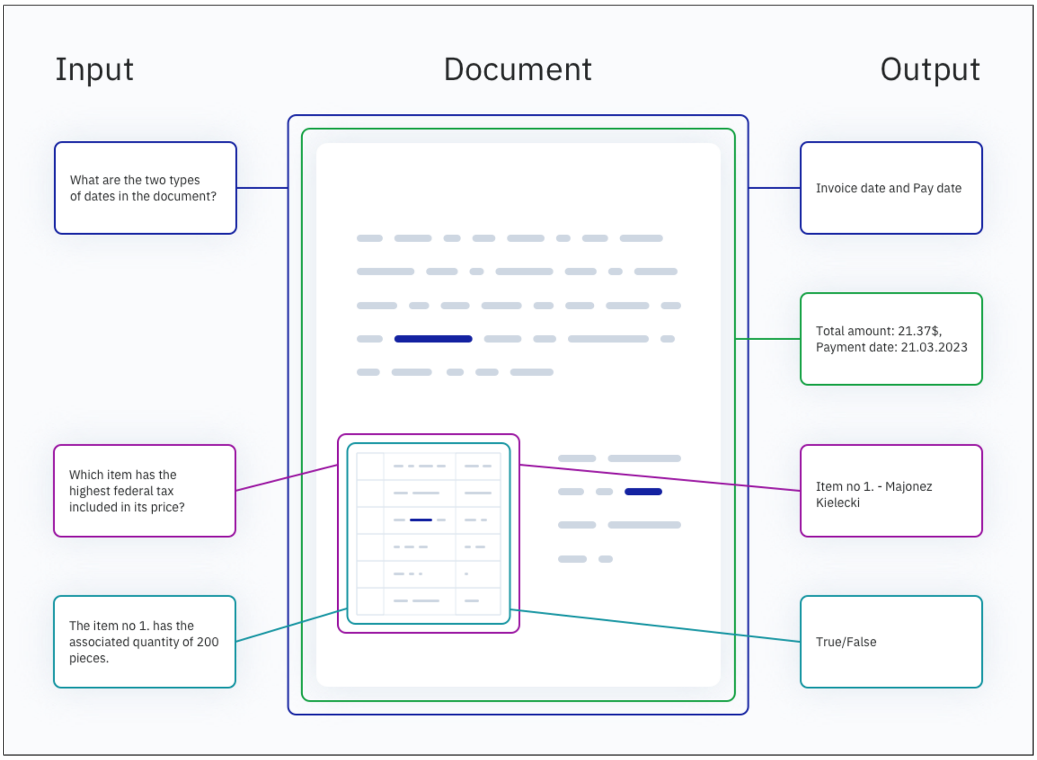

Input
Document
Output
What are the two types of dates in the document?
Invoice date and Pay date
Total amount: 21.37$, Payment date: 21.03.2023
Which item has the highest federal tax included in its price?
Item no 1. - Majonez Kielecki
The item no 1. has the associated quantity of 200 pieces.
True/False

Figure 1: Document Understanding covers problems ranging from the 🟩 extraction of key information, through 🟦 verification statements related to rich content, to 🟦 🟪 answering open questions regarding an entire file. It may involve the comprehension of multi-modal information conveyed by a document.

for freeing workers from paperwork through process automation is not yet addressed. Meanwhile, in other research communities, there are well-established progress measuring methods, like the most recognizable NLP benchmarks of GLUE and SuperGLUE covering a wide range of problems related to plain-text language understanding [50, 49] or VTAB and ImageNet in computer vision domain [56, 10]. We intend to bridge this major gap by introducing the first Document Understanding benchmark (available at `https://duebenchmark.com/`).

It includes tasks that either originally had a vital layout understanding component or were reformulated in such a way that after our modification they require layout understanding. In particular, there is no structured representation of the underlying text, such as a database-like table given in advance, and it has to be determined as a part of the end-to-end process from the input file. Every time, there is only a PDF file provided as an input with accompanying textual tokens and their locations (bounding boxes). It is not enough to process the text in a sequential manner (token by token), and there is no ground truth reading order given in advance.

Contribution. The idea of the paper is to gather, reformulate and unify a set of intuitively dissimilar tasks that we found to share the same underlying requirement of understanding layout concepts. In order to organize them in a useful benchmark, we contributed by performing the following steps:

1. We reviewed and selected the available datasets. Additionally, we reformulated three tasks to a document understanding setting and obtained original documents for all of them (PWC, WTQ, TabFact).

2. We performed data cleaning, including the improvements of data splits (DeepForm, WTQ), data deduplication, manual annotation (PWC, DeepForm), and converted data to a unified format (all datasets).

3. We implemented competitive baselines and measured human performance where it was required (PWC, DeepForm, WTQ).
4. We identified challenges related to the current progress in the DU domain's tasks and provided manually annotated diagnostic sets (all datasets).

These contributions are organized and described in Table 2. Additionally a wider search and review of available tasks is described in Appendix B.

2 The State of Document Understanding

We treat Document Understanding as an umbrella term covering problems of Key Information Extraction, Classification, Document Layout Analysis (DLA), Question Answering and Machine Reading Comprehension whenever they involve rich documents in contrast to plain texts or image-text pairs (Figure 1).

In addition to the problems strictly classified as Document Understanding, several related tasks can be reformulated as such. These provide either text-figure pairs instead of real-world documents or parsed tables given in their structured form. Since both can be rendered as synthetic documents with some loss of information involved, they are worth considering bearing in mind the low availability of proper Document Understanding tasks.

2.1 Landscape of Document Understanding Tasks

KIE. Key Information Extraction, also referred to as Property Extraction, is a task where tuple values of the form (property, document) are to be provided. Contrary to QA problems, there is no question in natural language but rather a phrase or keyword, such as *total amount*, or *place of birth*. Public datasets in the field include extraction performed on receipts [19, 35], invoices, reports [42], and forms [21]. Documents within each of the mentioned tasks are homogeneous, whereas the set of properties to extract is limited and known in advance – in particular, the same type-specific property names appear in both test and train sets. In contrast to Name Entity Recognition, KIE typically does not assume that token-level annotations are available, and may require normalization of values found within the document.

Classification. Though document image classification was initially approached using solely the methods of Computer Vision, it has recently become evident that multi-modal models can achieve significantly higher accuracy [52, 53, 37]. Similar conclusions were recently reached in other tasks, e.g., assigning labels to excerpts from biomedical papers [51].

DLA. Document Layout Analysis, performed to determine a document's components, was initially motivated by the need to optimize storage and the transmission of large information volumes [33]. Even though the motivation behind it has changed over the years, it is rarely an end in itself but rather a means to achieve a different goal, such as improving OCR systems. A typical dataset in the field assumes detection and classification of page regions or tokens [57, 27].

QA and MRC. At first glance, Question Answering and Machine Reading Comprehension over Documents is simply the KIE scenario where a question in natural language replaced a property name. More differences become evident when one notices that QA and MRC involve an open set of questions and various document types. Consequently, there is pressure to interpret the question and to possess better generalization abilities. Furthermore, a specific content to analyze demands a much stronger comprehension of visual aspects, as the questions commonly relate to figures and graphics accompanying the formatted text [30, 29, 46].

QA over figures. Question Answering over Figures is, to some extent, comparable with QA and MRC over documents described above. The difference is that a 'document' here consists of a single born-digital plot, reflecting information from chosen, desirably real-world data. Since questions in this category are typically templated and figures are synthetically generated by authors of the task, datasets in this category contain as much as millions of examples [31, 4].

QA and NLI over tables. Question Answering and Natural Language Inference over Tables are similar, though in the case of NLI, there is a statement to verify instead of a question to answer. There is never a need to analyze the actual layout, as both assume comprehension of a provided data structure in a way that is equivalent to a database table. Consequently, the methods proposed here are distinct from those used in Document Understanding [36, 6].

2.2 Gaps and Mistakes in Document Understanding Evaluation

Currently available datasets and previous work in the field cannot on their own provide enough information that would allow researchers to generalize results to other tasks within the Document Understanding paradigm. It is crucial to consider these tasks together, as they display a variety of characteristics a Document Understanding system may encounter in real-world applications. Notably, the scope of the challenges in a single dataset is limited to a specific task (e.g., Key Information Extraction, Question Answering) or to a particular (sub)problem (e.g., processing long documents in Kleister [42], layout understanding in DocBank [27]).

Simultaneously, a common practice in the community is to evaluate models on private data [24, 11, 34, 28] or task-specific datasets selected by authors independently [52, 53, 59, 37, 1, 18], making fair comparison difficult. Many publicly available datasets are too small to enable reliable comparison (FUNSD [21], Kleister NDA [42]) or are almost solved, i.e., there is no room for improvement due to annotation errors and near-perfect scores achieved by models nowadays (SROIE [20], CORD [35], RVL-CDIP [16]).

In light of the above circumstances, the review and selection of representative and reliable tasks is of great importance.

3 End-to-End Document Understanding Benchmark

The primary motivation for proposing this benchmark was to select datasets covering the broad range of tasks and DU-related problems satisfying the highest quality, difficulty, and licensing criteria.

Importantly, we opt for an end-to-end nature of tasks as opposed to, e.g., problems assuming some prior information on document layout. In particular, there is no structured representation of the underlying text, such as a database-like table given in advance, and it has to be determined from the raw input file as part of the end-to-end process.

We consider the aforementioned principle of end-to-end nature crucial because it ensures measurement to which degree manual workers can be supported in their repetitive tasks, i.e., how the ultimate goal of document understanding systems is supported in real-world applications. The said *alignment with real applications* is a vital characteristic of a good benchmark [26, 40].

3.1 Selected Datasets

Extensive documentation of the selection process, including the datasheet, is available in Appendices A-H and in the supplementary materials. Table 5 summarizes the selected tasks described in detail below, whereas Appendix B covers the complete list of considered datasets and reasons we omitted them.

Lack of the classification, layout analysis and figure QA tasks in this selection results from the fact that none of the available sets fulfills the assumed selection criteria.

The ★ symbol denotes that the dataset was reformulated or modified to improve its quality or align with the Document Understanding paradigm (see Table 2 and Appendix D). This symbol is not used to distinguish minor changes, such as data deduplication introduced in multiple datasets (Appendix C).

DocVQA. Dataset for Question Answering over single-page excerpts from various real-world industry documents. Typical questions present here might require comprehension of images, free text, tables, lists, forms, or their combination [30]. The best-performing solutions so far make use of layout-aware multi-modal models employing either encoder-decoder or sequence labeling architectures [37, 53].

InfographicsVQA. The task of answering questions about visualized data from a diverse collection of infographics, where the information needed to answer a question may be conveyed by text, plots, graphical or layout elements. Currently, the best result is obtained by an encoder-decoder model [29, 37].

Kleister Charity. A task for extracting information about charity organizations from their published reports is considered, as it is characterized by careful manual annotation by linguists and a significant gap to human performance. It addresses important areas, namely high layout variability (lack of templates), need for performing an OCR, the appearance of long documents, and multiple spatial features (e.g., tables, lists, and titles).

PWC★. Papers with Code Leaderboards dataset was designed to extract result tuples from machine learning papers, including information on task, dataset, metric name, score. The best performing approach involves a multi-step pipeline, with modules trained separately on identified subproblems [23]. In contrast to the original formulation, we provide a complete paper as input instead of the table. This approach allows us to treat the problem as an end-to-end Key Information Extraction task with grouped variables (Appendix D).

DeepForm★. KIE dataset consisting of socially important documents related to election spending. The task is to extract contract number, advertiser name, amount paid, and air dates from advertising disclosure forms submitted to the Federal Communications Commission [44]. We use a subset of distributed datasets and improve annotations errors and make the annotations between subsets for different years consistent (Appendix D).

WikiTableQuestions (WTQ)★. Dataset for QA over semi-structured HTML tables sourced from Wikipedia. The authors intended to provide complex questions, demanding multi-step reasoning on a series of entries in the given table, including comparison and arithmetic operations [36]. The problem is commonly approached assuming a semantic parsing paradigm, with an intermediate state of formal meaning representation, e.g., inferred query or predicted operand to apply on selected cells [55, 17]. We reformulate the task as document QA by rendering the original HTML and restrict available information to layout given by visible lines and token positions (Appendix D).

TabFact★. To study fact verification with semi-structured evidence over relatively clean and simple tables collected from Wikipedia, entailed and refuted statements corresponding to a single row or cell were prepared by the authors of TabFact [6]. Without being affected by the simplicity of binary classification, this task poses challenges due to the complex linguistic and symbolic reasoning required to perform with high accuracy. Analogously to WTQ, we render tables and reformulate the task as document NLI (Appendix D).

3.2 Diagnostic Subsets

As pointed out by Ruder, *to better understand the strengths and weaknesses of our models, we furthermore require more fine-grained evaluation* [40]. We propose several auxiliary validation subsets, spanning across all the tasks, to improve result analysis and aid the community in identifying where to focus its efforts. A detailed description of these categories and related annotation procedures is provided in Appendix G.

Answer characteristic. We consider four features regarding the shallow characteristic of the answer. First, we indicate whether the answer is provided in the text explicitly in exact form (*extractive* data point) or has to be inferred from the document content (*abstractive* one). The second category includes, e.g., all the cases where value requires normalization before being returned (e.g., changing

Table 1: Comparison of selected datasets with their base characteristic, including information regarding whether an input is an entire document (Doc.) or document excerpt (Exc.)

Task	Size (thousands)			Type	Metric	Features		Domain
	Train	Dev	Test			Input	Scanned	
DocVQA	10.2	1.3	1.3	Visual QA	ANLS	Doc.	+	Business
InfographicsVQA	4.4	.5	.6	Visual QA	ANLS		−	Open
Kleister Charity	1.7	.4	.6	KIE	F1		+/−	Legal
PWC★	.2	.06	.12	KIE*	F1		−	Scientific
DeepForm★	.7	.1	.3	KIE	F1		+/−	Finances
WikiTableQuestions★	1.4	.3	.4	Table QA	Acc.	Exc.	−	Open
TabFact★	13.2	1.7	1.7	Table NLI	Acc.		−	Open

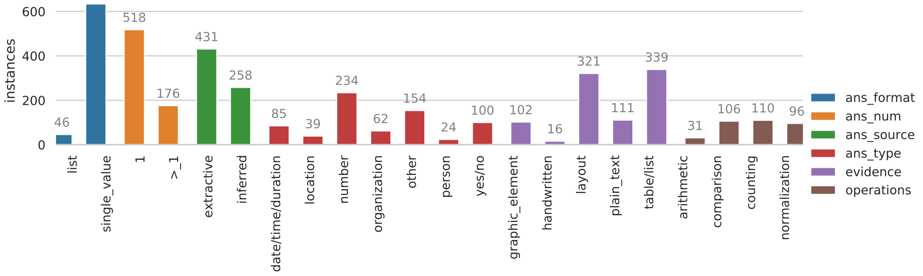


Figure 2: Number of annotated instances in each diagnostic subset category.

the date format). Next, we distinguish expected answers depending on whether they contain a *single value* or *list* of values. Finally, we decided to recognize several popular data types depending on shapes or class of expected named entity, i.e., to distinguish *date, number, yes/no, organization, location, and person* classes.

Evidence form. As we intend to analyze systems dealing with rich data, it is natural to study the performance w.r.t. the form that evidence is presented within the analyzed document. We distinguished *table/list, plain text, graphic element, layout,* and *handwritten* categories.

Required operation. Finally, we distinguish whether i.e., *arithmetic operation, counting, normalization* or some form of *comparison* has to be performed to answer correctly.

3.3 Intended Use

Data. We propose a unified data format for storing information in the Document Understanding domain and deliver converted datasets as part of the released benchmark (all selected datasets are hosted on the `https://duebenchmark.com/data` and can be downloaded from there). It assumes three interconnected dataset, document annotation and document content levels. The dataset level is intended for storing the general metadata, e.g., name, version, license, and source. The documents annotation level is intended to store annotations available for individual documents within datasets and related metadata (e.g., external identifiers). The content level store information about output and metadata from a particular OCR engine that was used to process documents (Appendix H).

Evaluation protocol. To evaluate a system on the DUE benchmark, one must create a JSON file with the results (in the data format mention above) based on the provided test data for each dataset and then upload all of the data to the website. Moreover, we establish a set of rules (Appendix I) which guarantees that all the benchmark submissions will be fair to compare, reproducible, and transparent (e.g., training performed on a development set is not allowed).

Leaderboard. We provide an online platform for the evaluation of Document Understanding models. To keep an objective means of comparison with the previously published results, we decided to retain

Table 2: Brief characteristics of our contribution, major fixes and modifications introduced to particular datasets. The enhancements of "Reformulation as DU" or "Improving data splits" are marked with ★ and are sufficient to consider the dataset unique; hence, achieved results are not comparable to the previously reported. See Appendix D for a full description of tasks processing.

Dataset	Diagnostic sets	Unified format	Human performance	Manual annotation	Reformulation as DU	Improved split
DocVQA	+	+	−	−	−	−
InfographicsVQA	+	+	−	−	−	−
Kleister Charity	+	+	−	−	−	−
PWC★	+	+	+	+	+	+
DeepForm★	+	+	+	+	−	+
WikiTableQuestions★	+	+	+	−	+	+
TabFact★	+	+	−	−	+	−

the initially formulated metrics. To calculate the global score we resort to an arithmetic mean of different metrics due to its simplicity and straightforward calculation.[2] In our platform we focus on customization, e.g., multiple leaderboards are available, and it is up to the participant to decide whether to evaluate the model on an entire benchmark or particular category. Moreover, we place attention to the explanation by providing means to analyze the performance concerning document or problem types (e.g., using the diagnostic sets we provide).[3]

4 Experiments

Following the evaluation protocol, the training is run three times for each configuration of model size, architecture, and OCR engine.

4.1 Baselines

The main focus of the experiments was to calculate baseline performance using a simple and popular model capable of solving all tasks without introducing any task-specific alterations. Employed methods were based on the previously released T5 model with a generic layout-modeling modification and pretraining.

T5. Text-to-text Transformer is particularly useful in studying performance on a variety of sequential tasks. We decided to rely on its extended version to identify the current level of performance on the chosen tasks and to facilitate future research by providing extensible architecture with a straightforward training procedure that can be applied to all of the proposed tasks in an end-to-end manner [38].

T5+2D. Extension of the model we propose assumes the introduction of 2D positional bias that has been shown to perform well on tasks that demand layout understanding [53, 37, 59]. We expect that comprehension of spatial relationships achieved in this way will be sufficient to demonstrate that methods from the plain-text NLP can be easily outperformed in the DUE benchmark.

Unsupervised pretraining. We constructed a corpus of documents with visually rich structure, based on 480k PDF files from the UCSF Industry Documents Library. It is used with a T5-like masked language model pretraining objective but in a salient span masking scheme where named entities are preferred over random tokens [38, 14]. An expected gain from its use is to tune 2D biases and become more robust to OCR errors and incorrect reading order.[4]

Human performance. We relied on the original estimation for DocVQA, InfographicsVQA, Charity, and TabFact datasets. For the PWC, WTQ and DeepForm estimation of human performance, we used the help of professional in-house annotators who are full-time employees of our company (see Appendix F). Each dataset was handled by two annotators; the average of their scores, when validated against the gold standard, is treated as the human performance (see Table 3). Interestingly, human scores on PWC are relatively low in terms of F1 value – we explained this and justified keeping the task in Appendix D.

4.2 Results

Comparison of the best-performing baselines to human performance and top results reported in the literature is presented in Table 3. In several cases, there is a small difference between the performance of our baselines and the external best. It can be attributed to several factors. First, the best results previously obtained on the tasks were task-specific, i.e., were explicitly designed for a particular task and did not support processing other datasets within the benchmark. Secondly, there are differences between the evaluation protocol that we assume and what the previous authors assumed (e.g., we do not allow training models on the development sets, we require reporting an average of multiple runs, we disallow pretraining on datasets that might lead to information leak). Thirdly, our baseline could not address examples demanding vision comprehension as it does not process image inputs. Finally,

[2] Scores on the DocVQA and InfographicsVQA test sets are calculated using the official website.

[3] We intend to gather datasets not included in the present version of the benchmark to facilitate evaluations in an entire field of DU, regardless of if they are included in the current version of the leaderboard.

[4] Details of the training procedure, such as used hyperparameters and source code, are available in the repository accompanying the paper.

Table 3: Best results of particular model configuration in relation to human performance and external best. The external bests marked with — were omitted due to the significant changes in the data sets. *U* stands for unsupervised pretraining.

Dataset / Task type	Score (task-specific metric)						
	T5	T5+2D	T5+U	T5+2D+U	External best		Human
DocVQA	72.5	74.1	76.4	81.3	87.1	[37]	98.1
InfographicsVQA	37.8	43.1	37.0	46.1	61.2	[37]	98.0
Kleister Charity	57.9	57.7	75.1	75.9	83.6	[59]	97.5
PWC★	24.2	25.2	25.1	27.3	—		51.1
DeepForm★	73.4	74.8	82.0	83.2	—		98.5
WikiTableQuestions★	32.5	33.4	38.1	44.0	—		76.7
TabFact★	52.2	53.7	67.9	70.6	—		92.1
Visual QA	55.2	58.6	56.7	63.7	n/a		98.1
KIE	51.8	52.6	60.7	62.1	n/a		82.4
Table QA/NLI	42.4	43.6	53.0	57.3	n/a		84.4
Overall	49.8	51.6	56.8	64.4	n/a		88.3

there is the case of Kleister Charity. An encoder-decoder model we relied on as a one-to-fit-all baseline cannot process an entire document due to memory limitations. As a result, the score was lower as we consumed only a part of the document. Note that external bests for reformulated tasks are no longer applicable to the benchmark in its present, more demanding form.

Irrespective of the task and whether our competitive baselines or external results are considered, there is still a large gap to humans, which is desired for novel baselines. Moreover, one can notice that the addition of 2D positional bias to the T5 architecture leads to better scores, which is yet another result we anticipated as it suggests that considered tasks have an essential component of layout comprehension.

Interestingly, the performance of the model can be significantly enhanced (up to 12.8 points difference) by providing additional data for unsupervised pretraining. Thus, the results not only support the premise that understanding 2D features demand more unlabeled data than the chosen datasets can offer but also lay a common ground between them, as the same layout-specific pretraining improved performance on all of them independently. This observation confirms that the notion of layout is a vital part of the chosen datasets.

4.3 Challenges of the Document Understanding Domain

Owing to its end-to-end nature and heterogeneity, Document Understanding is the touchstone of Machine Learning. However, the challenges begin to pile up due to the mere form a document is available in, as there is a widespread presence of analog materials such as scanned paper records. In the analysis below, we aim to explore the field of DU from the perspective of the model's development and point out the most critical limiting factors for achieving satisfying results.

Impact of OCR quality. We present detailed results for Azure CV and Tesseract OCR engine in Table 4. The differences in scores are huge for most of datasets (up to 18.4% in DocVQA) with clean advantage for Azure CV. Consequently, we see that architectures evaluated with different OCR engines are incomparable, e.g., the choice of an OCR engine may impact results more than the choice of model architecture. Moreover, with the usage of our diagnostic datasets we can observe that Tesseract struggle the most with *Handwritten* and *Table/list* categories in comparison to *Plain text* category. It is worth noting that we see a bigger difference in the results between Azure CV and Tesseract for *Extractive* category, which suggest us that we should used better OCR engine especially for that kind of problems.

Requirement of multi-modal comprehension. In addition to layout and textual semantics, part of the covered problems demand a Computer Vision component, e.g., to detect a logo, analyze a figure, recognize text style, determine whether the document was signed or the checkbox nearby was selected. Thus, Document Understanding naturally incorporates challenges of both multi-modality and each modality individually. Since none of our baselines contain a vision component, we underperform on

Table 4: Scores for different OCR engines and datasets with T5+2D model.

OCR	DocVQA	IVQA	Charity	DeepForm	Average	Average scores for different diagnostic categories				
						Extractive	Inferred	Handwritten	Table/list	Plain text
Azure CV (v3.2)	74.1	43.1	57.7	74.8	62.4	51.3	33.0	31.3	46.0	65.3
Tesseract (v4.0)	55.7	28.3	55.7	66.8	51.6	43.1	29.5	12.5	27.2	61.1

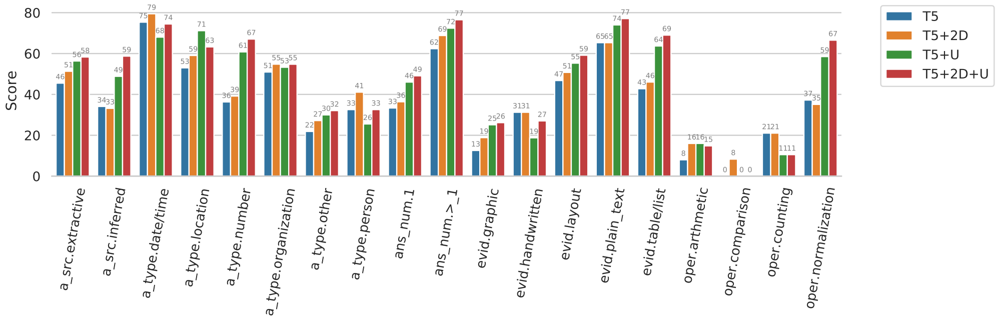


Figure 3: Results for diagnostic subsets. See Appendix G for detailed description of these categories.

the category of problems requiring multi-modality, as is visible on the diagnostic dataset we proposed. Nevertheless, better performance of the T5+2D model suggests that part of the problems considered as *visual*, can be in practice approximated by solely using the words' spatial relationships (e.g., text curved around a circle, located in the top-left corner of the page presumably has the logo inside).

Single architecture for all datasets. It is common that token-level annotation is not available, and one receives merely key-value or question-answer pairs assigned to the document. Even in problems of extractive nature, token spans cannot be easily obtained, and consequently, the application of state-of-the-art architectures from other tasks is not straightforward. In particular, authors attempting Document Understanding problems in sequence labeling paradigms were forced to rely on faulty handcrafted heuristics [37]. In the case of our baseline models, this problem is addressed straightforwardly by assuming a sequence-to-sequence paradigm that does not make use of token-level annotation. This solution, however, comes with a tradeoff of low performance on datasets requiring comprehension of long documents, such as Kleister Charity.

Diagnostic dataset. Our diagnostic datasets are an important part of the analysis of different challenges in general (e.g., OCR quality or multi-modal comprehension, as we mentioned above) and for debugging different types of architectural decisions (see Figure 3). For example, we can observe a big advantage of unsupervised pretraining in the *inferred, number, table/list* categories, which shows the importance of a good dataset for specific problems (dataset used for pretraining the original T5 model has a small number of documents containing tables). The most problematic categories for all models were those related to complex logic operations: *arithmetic, counting, comparison*.

5 Conclusions

To efficiently pass information to the reader, writers often assume that structured forms such as tables, graphs, or infographics are more accessible than sequential text due to human visual perception and our ability to understand a text's spatial surroundings. We investigate the problem of correctly measuring the progress of models able to comprehend such complex documents and propose a benchmark – a suite of tasks that balance factors such as quality of a document, importance of layout information, type and source of documents, task goal, and the potential usability in modern applications.

We aim to track the future progress on them with the website prepared for transparent verification and analysis of the results. The former is facilitated by the diagnostics subsets we derived to measure vital features of the Document Understanding systems. Finally, we provide a set of solid baselines, datasets in the unified format, and released source code to bootstrap the research on the topic.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See 2.1 where we discuss the broader landscape of available tasks and why we consider part of them.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Since this is a benchmark paper, we do not see any immediate negative societal impacts.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Please see Supplementary Materials.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Please see Appendix J and Supplementary Materials
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] Since we are providing baselines to roughly estimate whether the task is solved or not, or what type of information the documents contain, multiple runs are not neccessary.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix J.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See, e.g., Section 2.1.
   (b) Did you mention the license of the assets? [Yes] Only the datasets with permissive licenses were chosen, see Section 3.3.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Yes, we provide models, data and code.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] We annotated data by ourselves, based on instructions given in the Appendix G.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

References

[1] S. Appalaraju, B. Jasani, B. U. Kota, Y. Xie, and R. Manmatha. DocFormer: End-to-end transformer for document understanding, 2021.

[2] T. Bayer, J. Franke, U. Kressel, E. Mandler, M. Oberländer, and J. Schürmann. *Towards the Understanding of Printed Documents*, pages 3–35. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992.

[3] I. Chalkidis, E. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos. Large-scale multi-label text classification on EU legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6314–6322, Florence, Italy, July 2019. Association for Computational Linguistics.

[4] R. Chaudhry, S. Shekhar, U. Gupta, P. Maneriker, P. Bansal, and A. Joshi. Leaf-qa: Locate, encode attend for figure question answering. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3501–3510, 2020.

[5] L. Chen, X. Chen, Z. Zhao, D. Zhang, J. Ji, A. Luo, Y. Xiong, and K. Yu. WebSRC: A dataset for web-based structural reading comprehension, 2021.

[6] W. Chen, H. Wang, J. Chen, Y. Zhang, H. Wang, S. Li, X. Zhou, and W. Y. Wang. TabFact : A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, April 2020.

[7] W. Chen, H. Zha, Z. Chen, W. Xiong, H. Wang, and W. Wang. HybridQA: A dataset of multi-hop question answering over tabular and textual data, 2021.

[8] M. Cho, R. K. Amplayo, S. won Hwang, and J. Park. Adversarial TableQA: Attention supervision for question answering on tables, 2018.

[9] M. Dehghani. Toward document understanding for information retrieval. *SIGIR Forum*, 51(3):27–31, Feb. 2018.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[11] T. I. Denk and C. Reisswig. BERTgrid: Contextualized Embedding for 2D Document Representation and Understanding. In *Workshop on Document Intelligence at NeurIPS 2019*, 2019.

[12] F. Esposito, D. Malerba, G. Semeraro, and S. Ferilli. Knowledge revision for document understanding. In *ISMIS*, 1997.

[13] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. M. Wallach, H. D. III, and K. Crawford. Datasheets for datasets. *CoRR*, abs/1803.09010, 2018.

[14] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang. Retrieval augmented language model pre-training. In *ICML*, 2020.

[15] R. M. Haralick. Document image understanding: Geometric and logical layout. In *CVPR*, volume 94, pages 385–390, 1994.

[16] A. W. Harley, A. Ufkes, and K. G. Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2015.

[17] J. Herzig, P. K. Nowak, T. Müller, F. Piccinno, and J. Eisenschlos. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online, July 2020. Association for Computational Linguistics.

[18] T. Hong, D. Kim, M. Ji, W. Hwang, D. Nam, and S. Park. BROS: A layout-aware pre-trained language model for understanding documents, 2021.

[19] Z. Huang, K. Chen, J. He, X. Bai, D. Karatzas, S. Lu, and C. Jawahar. ICDAR2019 competition on scanned receipt OCR and information extraction. In *ICDAR*, 2019.

[20] Z. Huang, K. Chen, J. He, X. Bai, D. Karatzas, S. Lu, and C. V. Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520, 2019.

[21] G. Jaume, H. K. Ekenel, and J.-P. Thiran. FUNSD: A dataset for form understanding in noisy scanned documents, 2019.

[22] K. V. Jobin, A. Mondal, and C. V. Jawahar. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 1, pages 74–79, 2019.

[23] M. Kardas, P. Czapla, P. Stenetorp, S. Ruder, S. Riedel, R. Taylor, and R. Stojnic. AxCell: Automatic extraction of results from machine learning papers, 2020.

[24] A. R. Katti, C. Reisswig, C. Guder, S. Brarda, S. Bickel, J. Höhne, and J. B. Faddoul. Chargrid: Towards Understanding 2D Documents. *ArXiv*, abs/1809.08799, 2018.

[25] A. Kembhavi, M. Seo, D. Schwenk, J. Choi, A. Farhadi, and H. Hajishirzi. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. pages 5376–5384, 07 2017.

[26] S. Kounev, K. Lange, and J. von Kistowski. *Systems Benchmarking: For Scientists and Engineers*. Springer International Publishing, 2020.

[27] M. Li, Y. Xu, L. Cui, S. Huang, F. Wei, Z. Li, and M. Zhou. DocBank: A benchmark dataset for document layout analysis, 2020.

[28] B. P. Majumder, N. Potti, S. Tata, J. B. Wendt, Q. Zhao, and M. Najork. Representation learning for information extraction from form-like documents. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6495–6504, Online, July 2020. Association for Computational Linguistics.

[29] M. Mathew, V. Bagal, R. P. Tito, D. Karatzas, E. Valveny, and C. V. Jawahar. Infographicvqa, 2021.

[30] M. Mathew, D. Karatzas, and C. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2200–2209, January 2021.

[31] N. Methani, P. Ganguly, M. M. Khapra, and P. Kumar. PlotQA: Reasoning over scientific plots. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[32] L. Nan, C. Hsieh, Z. Mao, X. V. Lin, N. Verma, R. Zhang, W. Kryściński, N. Schoelkopf, R. Kong, X. Tang, M. Mutuma, B. Rosand, I. Trindade, R. Bandaru, J. Cunningham, C. Xiong, and D. Radev. FeTaQA: Free-form table question answering, 2021.

[33] D. Niyogi and S. N. Srihari. A rule-based system for document understanding. In *Proceedings of the Fifth AAAI National Conference on Artificial Intelligence*, pages 789–793, 1986.

[34] R. B. Palm, F. Laws, and O. Winther. Attend, copy, parse end-to-end information extraction from documents. *International Conference on Document Analysis and Recognition (ICDAR)*, 2019.

[35] S. Park, S. Shin, B. Lee, J. Lee, J. Surh, M. Seo, and H. Lee. CORD: A consolidated receipt dataset for post-ocr parsing. In *Document Intelligence Workshop at NeurIPS*, 2019.

[36] P. Pasupat and P. Liang. Compositional semantic parsing on semi-structured tables. *CoRR*, abs/1508.00305, 2015.

[37] R. Powalski, L. Borchmann, D. Jurkiewicz, T. Dwojak, M. Pietruszka, and G. Palka. Going Full-TILT boogie on document understanding with Text-Image-Layout Transformer. *CoRR*, abs/2102.09550, 2021.

[38] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text Transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[39] M. Rimol. Gartner Forecasts Worldwide Hyperautomation-Enabling Software Market to Reach Nearly $600 Billion by 2022. `https://www.gartner.com/en/newsroom/press-releases/2021-04-28-gartner-forecasts-worldwide-hyperautomation-enabling-software-market-to-reach-nearly-600-billion-by-2022`, 2021.

[40] S. Ruder. Challenges and Opportunities in NLP Benchmarking. `http://ruder.io/nlp-benchmarking`, 2021.

[41] Z. Shen, K. Lo, L. L. Wang, B. Kuehl, D. S. Weld, and D. Downey. Incorporating visual layout structures for scientific text classification, 2021.

[42] T. Stanisławek, F. Graliński, A. Wróblewska, D. Lipiński, A. Kaliska, P. Rosalska, B. Topolski, and P. Biecek. Kleister: Key information extraction datasets involving long documents with complex layouts, 2021.

[43] H. Sun, Z. Kuang, X. Yue, C. Lin, and W. Zhang. Spatial dual-modality graph reasoning for key information extraction, 2021.

[44] S. Svetlichnaya. DeepForm: Understand structured documents at scale. `https://wandb.ai/stacey/deepform_v1/reports/DeepForm-Understand-Structured-Documents-at-Scale--VmlldzoyODQ3Njg`, 2020.

[45] A. Talmor, O. Yoran, A. Catav, D. Lahav, Y. Wang, A. Asai, G. Ilharco, H. Hajishirzi, and J. Berant. Multimodalqa: Complex question answering over text, tables and images. *CoRR*, abs/2104.06039, 2021.

[46] R. Tanaka, K. Nishida, and S. Yoshida. VisualMRC: Machine reading comprehension on document images. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13878–13888, May 2021.

[47] S. L. Taylor, D. Dahl, M. Lipshutz, C. Weir, L. M. Norton, R. Nilson, and M. Linebarger. Integrated text and image understanding for document understanding. In *HLT*, 1994.

[48] H. M. Vu and D. T. Nguyen. Revising FUNSD dataset for key-value detection in document images. *CoRR*, abs/2010.05322, 2020.

[49] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[50] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics.

[51] T.-L. Wu, S. Singh, S. Paul, G. Burns, and N. Peng. MELINDA: A multimodal dataset for biomedical experiment method classification. *ArXiv*, abs/2012.09216, 2020.

[52] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou. LayoutLM: Pre-training of text and layout for document image understanding, 2019.

[53] Y. Xu, Y. Xu, T. Lv, L. Cui, F. Wei, G. Wang, Y. Lu, D. Florencio, C. Zhang, W. Che, M. Zhang, and L. Zhou. LayoutLMv2: Multi-modal pre-training for visually-rich document understanding, 2020.

[54] S. Yacoub. Automated quality assurance for document understanding systems. *IEEE Software*, 20(3):76–82, 2003.

[55] P. Yin, G. Neubig, W. tau Yih, and S. Riedel. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Annual Conference of the Association for Computational Linguistics (ACL)*, July 2020.

[56] X. Zhai, J. Puigcerver, A. Kolesnikov, P. Ruyssen, C. Riquelme, M. Lucic, J. Djolonga, A. S. Pinto, M. Neumann, A. Dosovitskiy, L. Beyer, O. Bachem, M. Tschannen, M. Michalski, O. Bousquet, S. Gelly, and N. Houlsby. A large-scale study of representation learning with the visual task adaptation benchmark, 2020.

[57] X. Zhong, J. Tang, and A. J. Yepes. PubLayNet: largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022. IEEE, Sep. 2019.

[58] F. Zhu, W. Lei, Y. Huang, C. Wang, S. Zhang, J. Lv, F. Feng, and T.-S. Chua. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3277–3287, Online, Aug. 2021. Association for Computational Linguistics.

[59] Łukasz Garncarek, R. Powalski, T. Stanisławek, B. Topolski, P. Halama, and F. Graliński. LAMBERT: Layout-aware (language) modeling using bert for information extraction, 2020.

A Issues Raised by Reviewers in the Previous Round

A.1 Major

The writing and organization of the paper confusing (pointed by Reviewer 2S9y, 8jTw and Program Chairs)

The paper was reorganized and extended where needed. In particular, we have rewritten and explained crucial aspects such as what the selected tasks have in common, why such a benchmark is necessary, and what its relation is to other benchmarks. Additionally, we described the contribution in detail and changed the sections' order to make the reasoning easier to comprehend. The experimental section and discussion of results were substantially extended. Several parts were shortened, and we elaborated on the benchmark aspect that previously was unclear to the reviewers.

The experiments section should be improved (pointed by Reviewer 2S9y, 8jTw and Program Chairs) — not enough baseline methods and in-depth discussions

We substantially improved the baselines by introducing a stage of unsupervised pretraining on visually rich documents. Additionally, we provided an extensive evaluation concerning the challenges (Section 4.3) with the use of diagnostic sets we introduced. Significantly, we commented on the results and clarified why they could not be compared to those obtained by other authors, i.e., that the evaluation rules we assume exclude the previous submission, or they were obtained on a task before its reformulation to document understanding.

Not well justified why such such "document understanding" benchmark is needed (pointed by Reviewer 2S9y, 8jTw and Program Chairs)

It is now clarified in the introduction, Section 3, and, especially, Section 2.2. Additionally, it is shown that all the document understanding tasks share the same challenges and thus are worth a parallel investigation in Section 4.3.

The main contribution of this paper is unclear (pointed by Reviewer 2S9y, 8jTw)

It is now stated explicitly in the introduction, where the Contribution section was added.

Relation To Prior Work. The paper does not mention the difference between this work and previous ones. (pointed by Reviewer 2S9y, 8jTw))

We clarify the difference and refer to the previous works in the Section 2.2.

A.2 Minor

It would be helpful if the authors could explain more on why these 7 datasets should be considered together. What is it about these tasks that fundamentally suggests they should be treated together? The authors allude to the idea that in all the these tasks, the model must comprehend the "layout" of the documents. A bit more precision and exposition here would be helpful. (Reviewer 2S9y)

Similarly to the motivation behind the benchmark, it is now addressed in Section 3, Section 2.2, and Section 4.3.

The details are partially provided. Specifically, the authors describe the format of data organisation, but lack the other aspects like maintenance, ethics, responsibility, etc. (Reviewer 8jTw)

To answer the call for an explicit statement about hosting, licensing, and maintenance, we updated our supplementary material with the filled data sheet based on Gebru et. al *Datasheets for Datasets* article, where such information is detailed (filling this form is required for datasets in this track, but we adapted it to the benchmark).

B Considered datasets

B.1 Desired characteristics

End-to-end nature. As the value and importance of Document Understanding result from its application to process automation, a good benchmark should measure to which degree workers can be supported in their tasks. Though Layout Analysis is oldest of the Document Understanding problems, its output is often not an end in itself but rather a half-measure disconnected from the final information the system is used for. We also remove all tasks which as an input takes collection of documents.

Quality. Availability of high-quality annotation was a condition *sine qua non* for a task to qualify. To ensure the highest annotation quality, we excluded resources prepared using a distant annotation procedure, e.g., classification tasks where entire sources were labeled instead of individual instances, or templated question-answer pairs.

Difficulty. As it makes no sense to measure progress on solved problems, only tasks with a substantial gap between human performance and state-of-the-art models were considered. In the case of promising tasks lacking a human baseline, we provided our estimation. Moreover, we remove all tasks were free text was dominated in documents (we don't need to use layout or visual features).

Licensing. In publishing our benchmark, we are making efforts to ensure the highest standards for the future of the machine learning community. Only tasks with a permissive license to use annotations and data for further research can be considered.

At the same time, we recognized it is essential to approach the benchmark construction holistically, i.e., to carefully select tasks from diverse domains and types in the rare cases where datasets are abundant.

B.2 Datasets selection process

The review protocol consisted of a manual search in specific databases, repositories and distribution services. The scientific resources included in the search were:

- `https://paperswithcode.com/datasets/`
- `https://datasetsearch.research.google.com/`
- `https://data.mendeley.com/`
- `https://arxiv.org/search/`
- `https://github.com/`
- `https://allenai.org/data/`
- `https://www.semanticscholar.org/`
- `https://scholar.google.com/`
- `https://academic.microsoft.com/home`

Results were reviewed by one of authors of the present paper and the resources related to classification, KIE, QA, MRC, and NLI over complex documents, figures, and tables were identified as potentially relevant (in accordance with inclusion criteria described in Section B.2).

The initial search assumed use of the following keywords: *Question Answering, Visual Question Answering, Document Question Answering, Document Classification, Document Dataset, Information Extraction*. Additionally, we used *Machine Reading Comprehension, Question Answering, VQA* in combination with *Document*, and *Visual, Document, Table, Figure, Plot, Chart, Hybrid* in combination with *Question Answering* or *Information Extraction*.

Table 5 presents list of relevant datasets and results of their assessment according to the criteria of end-to-end nature, quality, difficulty, and licensing. Candidate tasks resulted from an extensive review of both literature and data science challenges without accompanying publication and their basic characteristics.

Table 5: Comparison of selected and considered datasets with their base characteristic, including information regarding whether an input is a collection of documents (Col.), entire document (Doc.) or document excerpt (Exc.).

Dataset	Type	Size (thousands)			Selection criteria				Input	Domain	Comment
		Train	Dev	Test	End-to-end	Quality	Difficulty	Licensing			
Kleister Charity [42]	KIE	1.73	.44	.61	+	+	+	+	Doc.	Finances	
PWC [23]	KIE	.2	.06	.12	+	+	+	+	Doc.	Scientific	
DeepForm [44]	KIE	.7	.1	.3	+	+	+	+	Doc.	Finances	
DocVQA [30]	Visual QA	10.2	1.3	1.3	+	+	+	+	Doc.	Business	
InfographicsVQA [29]	Visual QA	4.4	.5	.6	+	+	+	+	Doc.	Open	
TabFact [6]	Table NLI	13.2	1.7	1.7	+	+	+	+	Exc.	Open	
WTQ [36]	Table QA	1.4	.3	.4	+	+	+	+	Exc.	Open	
Kleister NDA [42]	KIE	.25	.08	.2	+	+	−	+	Doc.	Legal	Dominated by extraction from free text
SROIE [19]	KIE	.63	-	.35	+	+	−	+	Doc.	Finances	No room for improvement
CORD [35]	KIE	.8	.1	.1	+	+	−	+	Doc.	Finances	No room for improvement
Wildreceipt [43]	KIE	1.27	-	.47	+	+	−	+	Doc.	Finances	No room for improvement
WebSRC [5]	KIE	4.55	.9	1.0	+	−	+	+	Doc.	Open	Templated input data
FUNSD [21]	KIE	.15	-	.05	+	−	+	+	Doc.	Finances	Known disadvantages [48]
DocCVQA [29]	Visual QA	4.4	.5	.6	−	+	+	+	Col.	Open	Document Collection Question Answering
TextbookQA [25]	Visual QA	.67	.2	.21	+	−	+	+	Doc.	Educational	Source files are not available
MultiModalQA [45]	Visual QA	23.82	2.44	3.66	+	−	+	+	Doc.	Open	Automatically generated questions
VisualMRC [46]	Visual MRC	7	1	2	+	+	−	+	Doc.	Open	Human performance reached
RVL-CDIP [16]	Classification	320	40	40	+	+	−	+	Doc.	Finances	No room for improvement
DocFigure [22]	Classification	19.8	-	13.1	+	+	−	+	Doc.	Scientific	No room for improvement
EURLEX57K [3]	Classification	45	6	6	+	+	−	+	Doc.	Legal	Dominated by extraction from free text
MELINDA [51]	Classification	4.34	.45	.58	+	−	+	+	Doc.	Scientific	Semi-supervised annotation
S2-VL [41]	DLA	1.3	-	-	−	+	+	+	Doc.	Scientific	Cross-validation for training and testing
DocBank [27]	DLA	398	50	50	−	−	+	+	Doc.	Scientific	Automatic annotation
Publaynet [57]	DLA	340.4	11.9	12	−	−	+	+	Doc.	Scientific	Automatic annotation
PlotQA [31]	Figure QA	157	33.7	33.7	+	−	+	+	Exc.	Open	Synthetic
Leaf-QA [4]	Figure QA	200	40	8.15	+	−	+	+	Exc.	Open	Templated questions
TAT-QA [58]	Table QA	2.2	.28	.28	+	−	+	+	Exc.	Finances	Source files are not available
WikiOPS [8]	Table QA	17.28	2.47	4.67	+	+	−	+	Exc.	Open	No room for improvement
FeTaQA [32]	Table QA	7.33	1.0	2.0	+	−	+	+	Exc.	Open	Answers as a free-form text
HybridQA [7]	Table QA	62.68	3.47	3.46	−	+	+	+	Col.	Open	Multihop Question Answering

C Minor dataset modifications

Deduplication. Through the systematic analysis and validation of the chosen datasets, we noticed one of the commonly appearing defects is the presence of duplicated annotations. We decided to remove these duplicates from InfographicsVQA (14 annotations from train, two from the dev set), DocVQA (four from train and test sets each), TabFact (309 from train, 53 from dev, and 52 the test set), and WikiTableQuestions (one annotation from each train and test sets).

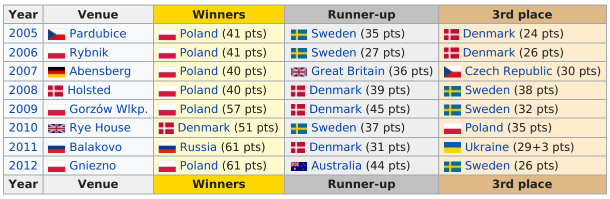
D Tasks processing and reformulation

Since part of the datasets were reformulated or modified to improve the benchmark quality or align the task with the Document Understanding paradigm, we describe the introduced changes in detail below.

WikiTableQuestions*. We prepare input documents by rendering table-related HTML distributed by authors in *wkhtmltopdf* and crop the resulting files with *pdfcrop*. As these code excerpts do not contain *head* tag with JavaScript and stylesheet references, we use the header from the present version of the Wikipedia website.

Approximately 10% of tables contained at least one *img* tag with a source that is no longer reachable. It results in a question mark icon displayed instead of the image and does not impact the evaluation procedure since the questions here do not require image comprehension.

The original WTQ dataset consists of *training*, *pristine-seen-tables*, and *pristine-unseen-tables* subsets. We treat *pristine-unseen-tables* as a test set and create new training and development sets by rearranging data from *training* and *pristine-seen-tables*. The latter operation is dictated by the leakage of documents in the original formulation, i.e., we consider it undesirable for a document to appear in different splits, even if the question differs. The resulting dataset consists of approximately 2100 documents divided in the proportion of 65%, 15%, 20% into training, development, and test sets.

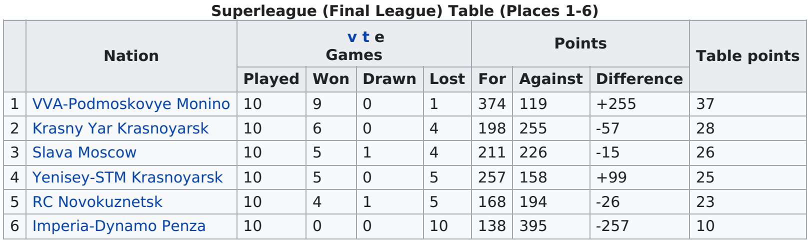


Year	Venue	Winners	Runner-up	3rd place
2005	Pardubice	Poland (41 pts)	Sweden (35 pts)	Denmark (24 pts)
2006	Rybnik	Poland (41 pts)	Sweden (27 pts)	Denmark (26 pts)
2007	Abensberg	Poland (40 pts)	Great Britain (36 pts)	Czech Republic (30 pts)
2008	Holsted	Poland (40 pts)	Denmark (39 pts)	Sweden (38 pts)
2009	Gorzów Wlkp.	Poland (57 pts)	Denmark (45 pts)	Sweden (32 pts)
2010	Rye House	Denmark (51 pts)	Sweden (37 pts)	Poland (35 pts)
2011	Balakovo	Russia (61 pts)	Denmark (31 pts)	Ukraine (29+3 pts)
2012	Gniezno	Poland (61 pts)	Australia (44 pts)	Sweden (26 pts)
Year	Venue	Winners	Runner-up	3rd place

Figure 4: Document in WikiTableQuestions reformulated as Document Understanding.

(Question) After their first place win in 2009, how did Poland place the next year at the speedway junior world championship? (Answer) 3rd place

TabFact★. As the authors of TabFact distribute only CSV files, we resorted to HTML from the WikiTables dump their CSV were presumably generated from.[5] As Chen et al. [6] dropped some of the columns present in used WikiTable tables, we remove them too, to ensure compatibility with the original TabFact. Rendered files are used analogously to the case of WTQ.



Superleague (Final League) Table (Places 1-6)

	Nation	v t e Games				Points			Table points
		Played	Won	Drawn	Lost	For	Against	Difference	
1	VVA-Podmoskovye Monino	10	9	0	1	374	119	+255	37
2	Krasny Yar Krasnoyarsk	10	6	0	4	198	255	-57	28
3	Slava Moscow	10	5	1	4	211	226	-15	26
4	Yenisey-STM Krasnoyarsk	10	5	0	5	257	158	+99	25
5	RC Novokuznetsk	10	4	1	5	168	194	-26	23
6	Imperia-Dynamo Penza	10	0	0	10	138	395	-257	10

Figure 5: Document in TabFact reformulated as Document Understanding.

(Claim) To calculate table point, a win be worth 3, a tie be worth 1 and a loss be worth 0

Results differ from TabFact in several aspects, i.e., text in our variant is not normalized, it includes the original formatting, and the tables are more complex due to restoring the original cell merges. All mentioned differences are desired, as we intended to consider raw, unprocessed files without any heuristics or normalization applied.

Another difference we noticed is that tables in the original TabFact are sometimes one row shorter, i.e., they do not contain the last row present in the WikiTable dump. As it should not impact expected answers, we decided to maintain the fidelity to Wikipedia and use the complete table.

We use the original splits into training, development, and test sets.

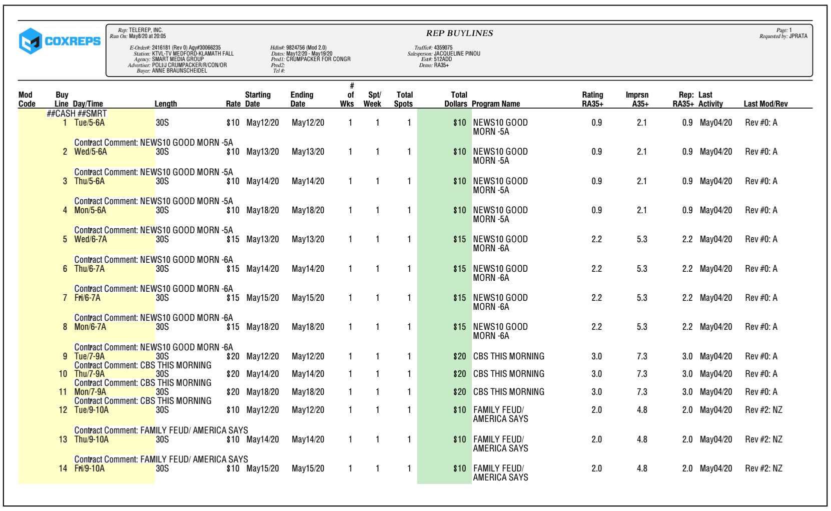
DeepForm★. The original DeepForm dataset consists of 2012, 2014, and 2020 subsets differing in terms of annotation quality and documents' diversity. We decided to use only the 2020 subset as for 2014, and 2020 annotations were prepared either automatically or by volunteers, leading to questionable quality. The selected subset was randomly divided into training, development and test set.

We noticed several inconsistencies during the initial analysis that lead us to the manual correction of autodetected: (1) invalid date format; (2) flight start dates earlier than flight end; (3) documents lacking one or more data points.

In addition to the improved 2020 subset, we manually annotated one hundred 2012 documents, as they can pose different challenges (contain different document templates, handwriting, have lower image quality). They were used to extend development and test set. The final dataset consists of 700 training, 100 development, and 300 test set documents.

PWC★. The authors of AxCell relied on PWC Leaderboards and LinkedResults datasets [23]. The original formulation assumes extraction of *(task, dataset, metric, model, score)* tuples from

[5] `http://websail-fe.cs.northwestern.edu/TabEL/tables.json.gz`



COXREPS

Rep: TELEREP, INC.
Run On: May8/20 at 20:05

REP BUYLINES

Page: 1
Requested by: JPRATA

E-Order#: 2416181 (Rev 0) Agy#30066235
Station: KTVL-TV MEDFORD-KLAMATH FALL
Agency: SMART MEDIA GROUP
Advertiser: POLI/J CRUMPACKER/R/CON/OR
Buyer: ANNE BRAUNSCHEIDEL

Hdln#: 9824756 (Mod 2.0)
Dates: May12/20 - May19/20
Prod1: CRUMPACKER FOR CONGR
Prod2:
Tel #:

Traffic#: 4359075
Salesperson: JACQUELINE PINOU
Est#: 512ADD
Demo: RA35+

Mod Code	Buy Line	Day/Time	Length	Rate	Starting Date	Ending Date	# of Wks	Spt/ Week	Total Spots	Total Dollars	Program Name	Rating RA35+	Imprsn A35+	Rep: RA35+	Last Activity	Last Mod/Rev
	##CASH ##SMRT															
	1	Tue/5-6A	30S	$10	May12/20	May12/20	1	1	1	$10	NEWS10 GOOD MORN -5A	0.9	2.1	0.9	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -5A														
	2	Wed/5-6A	30S	$10	May13/20	May13/20	1	1	1	$10	NEWS10 GOOD MORN -5A	0.9	2.1	0.9	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -5A														
	3	Thu/5-6A	30S	$10	May14/20	May14/20	1	1	1	$10	NEWS10 GOOD MORN -5A	0.9	2.1	0.9	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -5A														
	4	Mon/5-6A	30S	$10	May18/20	May18/20	1	1	1	$10	NEWS10 GOOD MORN -5A	0.9	2.1	0.9	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -5A														
	5	Wed/6-7A	30S	$15	May13/20	May13/20	1	1	1	$15	NEWS10 GOOD MORN -6A	2.2	5.3	2.2	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -6A														
	6	Thu/6-7A	30S	$15	May14/20	May14/20	1	1	1	$15	NEWS10 GOOD MORN -6A	2.2	5.3	2.2	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -6A														
	7	Fri/6-7A	30S	$15	May15/20	May15/20	1	1	1	$15	NEWS10 GOOD MORN -6A	2.2	5.3	2.2	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -6A														
	8	Mon/6-7A	30S	$15	May18/20	May18/20	1	1	1	$15	NEWS10 GOOD MORN -6A	2.2	5.3	2.2	May04/20	Rev #0: A
		Contract Comment: NEWS10 GOOD MORN -6A														
	9	Tue/7-9A	30S	$20	May12/20	May12/20	1	1	1	$20	CBS THIS MORNING	3.0	7.3	3.0	May04/20	Rev #0: A
		Contract Comment: CBS THIS MORNING														
	10	Thu/7-9A	30S	$20	May14/20	May14/20	1	1	1	$20	CBS THIS MORNING	3.0	7.3	3.0	May04/20	Rev #0: A
		Contract Comment: CBS THIS MORNING														
	11	Mon/7-9A	30S	$20	May18/20	May18/20	1	1	1	$20	CBS THIS MORNING	3.0	7.3	3.0	May04/20	Rev #0: A
		Contract Comment: CBS THIS MORNING														
	12	Tue/9-10A	30S	$10	May12/20	May12/20	1	1	1	$10	FAMILY FEUD/ AMERICA SAYS	2.0	4.8	2.0	May04/20	Rev #2: NZ
		Contract Comment: FAMILY FEUD/ AMERICA SAYS														
	13	Thu/9-10A	30S	$10	May14/20	May14/20	1	1	1	$10	FAMILY FEUD/ AMERICA SAYS	2.0	4.8	2.0	May04/20	Rev #2: NZ
		Contract Comment: FAMILY FEUD/ AMERICA SAYS														
	14	Fri/9-10A	30S	$10	May15/20	May15/20	1	1	1	$10	FAMILY FEUD/ AMERICA SAYS	2.0	4.8	2.0	May04/20	Rev #2: NZ

Figure 6: Single page from document in DeepForm.

a provided table. In contrast, we reformulate the task as Document Understanding and provide a complete paper as input instead. These are obtained using arXiv identifiers available in the PWC metadata. Consequently, the resulting task is an end-to-end Key Information Extraction from real-world scientific documents.

Whereas LinkedResults was annotated consistently, the PWC is of questionable quality as it was obtained from leaderboards filled by Papers with Code visitors without a clear guideline or annotation rules. The difference between the two is substantial, i.e., the agreement in terms of F1 score between publications present in both PWC and LinkedResults is lower than 0.35. We attribute this mainly to flaws in the PWC dataset, such as missing records, inconsistent normalization and the difficulty of the task itself.

Consequently, we decided to perform its manual re-annotation assuming that: (1) The best result for a proposed model variant on the single dataset has to be annotated, e.g., if two models with different parameter sizes were present in the table, we report only the best one. (2) Single number is preferred (we take the average over multiple split or parts of the dataset if possible). (3) When results from the test set are available, we prefer them and don't report results from the validation set. (4) We add multiple value variants when possible. (5) We include information on used validation/dev/test split in the dataset description wherever applicable. (6) We don't report results on the train set. (7) We don't annotate results not appearing in the table. (8) We filter out publications that are hard to annotate even for a human.

Interestingly, human scores on PWC are relatively low in terms of F1 value. This can be attributed to unrestricted nature of particular properties, e.g., *accuracy* and *average accuracy* are equally valid metric values. Similarly, *Action Recognition*, *Action Classification*, and *Action Recognition* are equally valid task names. At the same time, it is impossible to provide all answer variants during the preparation of the gold standard. We decided to keep the dataset in the benchmark as it is extremely demanding, and there is still a large gap between humans' and models' performance (See Table 3).

E Dataset statistics

Chosen datasets represent the plethora of domains, lengths, and document types. This appendix covers the critical aspects of particular tasks at the population level.

Though part of the datasets is limited to one-pagers, the remaining documents range from a few to few hundred pages (Figure 8). At the same time, there is a great variety in how much text is present on a single page – we have both densely packed scientific documents and concise document excerpts or infographics. This diversity allows us to measure the ability to comprehend documents depending on their length.

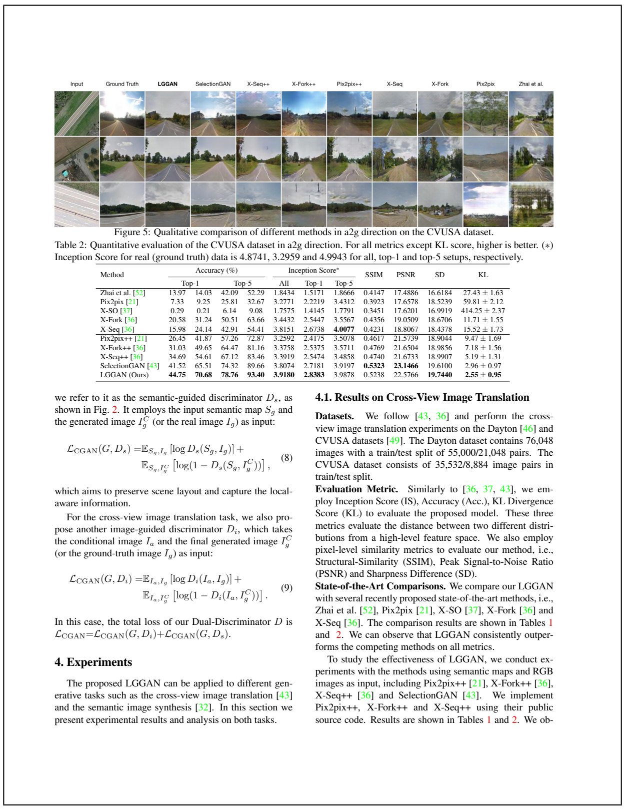


Figure 5: Qualitative comparison of different methods in a2g direction on the CVUSA dataset.

Table 2: Quantitative evaluation of the CVUSA dataset in a2g direction. For all metrics except KL score, higher is better. (∗) Inception Score for real (ground truth) data is 4.8741, 3.2959 and 4.9943 for all, top-1 and top-5 setups, respectively.

Method	Accuracy (%)				Inception Score*			SSIM	PSNR	SD	KL
	Top-1		Top-5		All	Top-1	Top-5				
Zhai et al. [52]	13.97	14.03	42.09	52.29	1.8434	1.5171	1.8666	0.4147	17.4886	16.6184	27.43 ± 1.63
Pix2pix [21]	7.33	9.25	25.81	32.67	3.2771	2.2219	3.4312	0.3923	17.6578	18.5239	59.81 ± 2.12
X-SO [37]	0.29	0.21	6.14	9.08	1.7575	1.4145	1.7791	0.3451	17.6201	16.9919	414.25 ± 2.37
X-Fork [36]	20.58	31.24	50.51	63.66	3.4432	2.5447	3.5567	0.4356	19.0509	18.6706	11.71 ± 1.55
X-Seq [36]	15.98	24.14	42.91	54.41	3.8151	2.6738	**4.0077**	0.4231	18.8067	18.4378	15.52 ± 1.73
Pix2pix++ [21]	26.45	41.87	57.26	72.87	3.2592	2.4175	3.5078	0.4617	21.5739	18.9044	9.47 ± 1.69
X-Fork++ [36]	31.03	49.65	64.47	81.16	3.3758	2.5375	3.5711	0.4769	21.6504	18.9856	7.18 ± 1.56
X-Seq++ [36]	34.69	54.61	67.12	83.46	3.3919	2.5474	3.4858	0.4740	21.6733	18.9907	5.19 ± 1.31
SelectionGAN [43]	41.52	65.51	74.32	89.66	3.8074	2.7181	3.9197	**0.5323**	**23.1466**	19.6100	2.96 ± 0.97
LGGAN (Ours)	**44.75**	**70.68**	**78.76**	**93.40**	**3.9180**	**2.8383**	3.9878	0.5238	22.5766	**19.7440**	**2.55 ± 0.95**

we refer to it as the semantic-guided discriminator D_s, as shown in Fig. 2. It employs the input semantic map S_g and the generated image I_g^C (or the real image I_g) as input:

$$\begin{aligned}\mathcal{L}_{\mathrm{CGAN}}(G, D_s) =&\mathbb{E}_{S_g,I_g}\left[\log D_s(S_g, I_g)\right] + \\ &\mathbb{E}_{S_g,I_g^C}\left[\log(1 - D_s(S_g, I_g^C))\right],\end{aligned} \tag{8}$$

which aims to preserve scene layout and capture the local-aware information.

For the cross-view image translation task, we also propose another image-guided discriminator D_i, which takes the conditional image I_a and the final generated image I_g^C (or the ground-truth image I_g) as input:

$$\begin{aligned}\mathcal{L}_{\mathrm{CGAN}}(G, D_i) =&\mathbb{E}_{I_a,I_g}\left[\log D_i(I_a, I_g)\right] + \\ &\mathbb{E}_{I_a,I_g^C}\left[\log(1 - D_i(I_a, I_g^C))\right].\end{aligned} \tag{9}$$

In this case, the total loss of our Dual-Discriminator D is $\mathcal{L}_{\mathrm{CGAN}}$=$\mathcal{L}_{\mathrm{CGAN}}(G, D_i)$+$\mathcal{L}_{\mathrm{CGAN}}(G, D_s)$.

4. Experiments

The proposed LGGAN can be applied to different generative tasks such as the cross-view image translation [43] and the semantic image synthesis [32]. In this section we present experimental results and analysis on both tasks.

4.1. Results on Cross-View Image Translation

Datasets. We follow [43, 36] and perform the cross-view image translation experiments on the Dayton [46] and CVUSA datasets [49]. The Dayton dataset contains 76,048 images with a train/test split of 55,000/21,048 pairs. The CVUSA dataset consists of 35,532/8,884 image pairs in train/test split.

Evaluation Metric. Similarly to [36, 37, 43], we employ Inception Score (IS), Accuracy (Acc.), KL Divergence Score (KL) to evaluate the proposed model. These three metrics evaluate the distance between two different distributions from a high-level feature space. We also employ pixel-level similarity metrics to evaluate our method, i.e., Structural-Similarity (SSIM), Peak Signal-to-Noise Ratio (PSNR) and Sharpness Difference (SD).

State-of-the-Art Comparisons. We compare our LGGAN with several recently proposed state-of-the-art methods, i.e., Zhai et al. [52], Pix2pix [21], X-SO [37], X-Fork [36] and X-Seq [36]. The comparison results are shown in Tables 1 and 2. We can observe that LGGAN consistently outperforms the competing methods on all metrics.

To study the effectiveness of LGGAN, we conduct experiments with the methods using semantic maps and RGB images as input, including Pix2pix++ [21], X-Fork++ [36], X-Seq++ [36] and SelectionGAN [43]. We implement Pix2pix++, X-Fork++ and X-Seq++ using their public source code. Results are shown in Tables 1 and 2. We ob-

Figure 7: Single page from document in PWC.

F Details of human performance estimation

Estimation of human performance for PWC, WikiTableQuestions, DeepForm was performed in-house by professional annotators who are full-time employees of Applica.ai. Before approaching the process, each of them has to participate in the task-specific training described below.

Number of annotated samples depended on task difficulty and the variance of the resulting scores. We relied on 50 fully annotated papers for the PWC dataset (approx. 150 tuples with five values each), 109 DeepForm documents (532 values), and 300 questions asked to different WikiTableQuestion tables.

Each dataset was approached with two annotators in the LabelStudio tool. Human performance is the average of their scores when validated against the gold standard.

Training. Each person participating in the annotation process completed the training consisting of four stages: (1) Annotation of five random documents from the task-specific development set. (2) Comparative analysis of differences between their annotations and the gold standard. (3) Annotation of ten random documents from the task-specific development set and subsequent comparative analysis. (4) Discussion between annotators aimed at agreeing on the shared, coherent annotation rules.

G Annotation of diagnostic subsets

In order to analyze the prepared benchmark and the results of individual models, diagnostic sets were prepared. These diagnostic sets are subsets of examples selected from the testset for all datasets.

When building a taxonomy for diagnostic sets, we adopted two basic assumptions: (1) It must be consistent across all selected tasks so that at least two tasks can be noted with a given category (2)

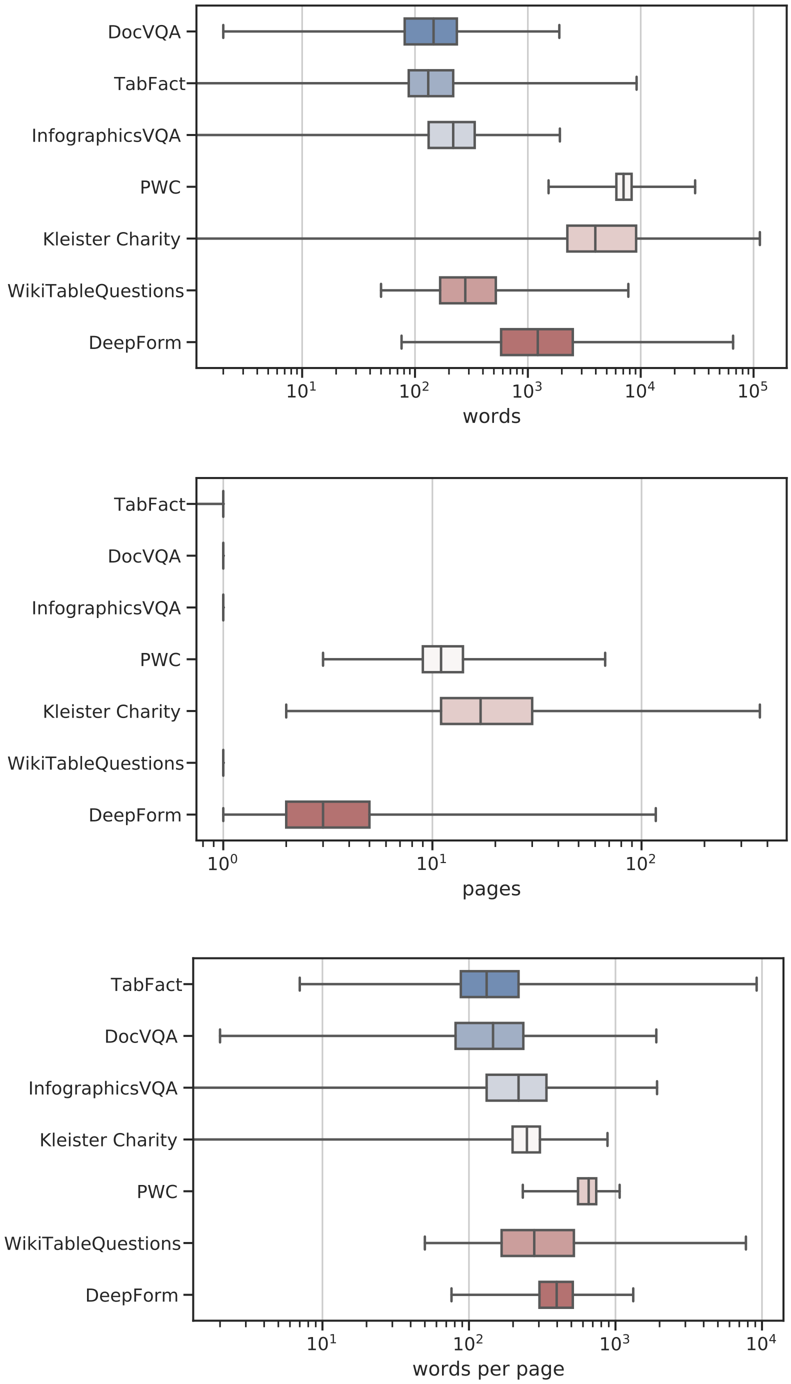
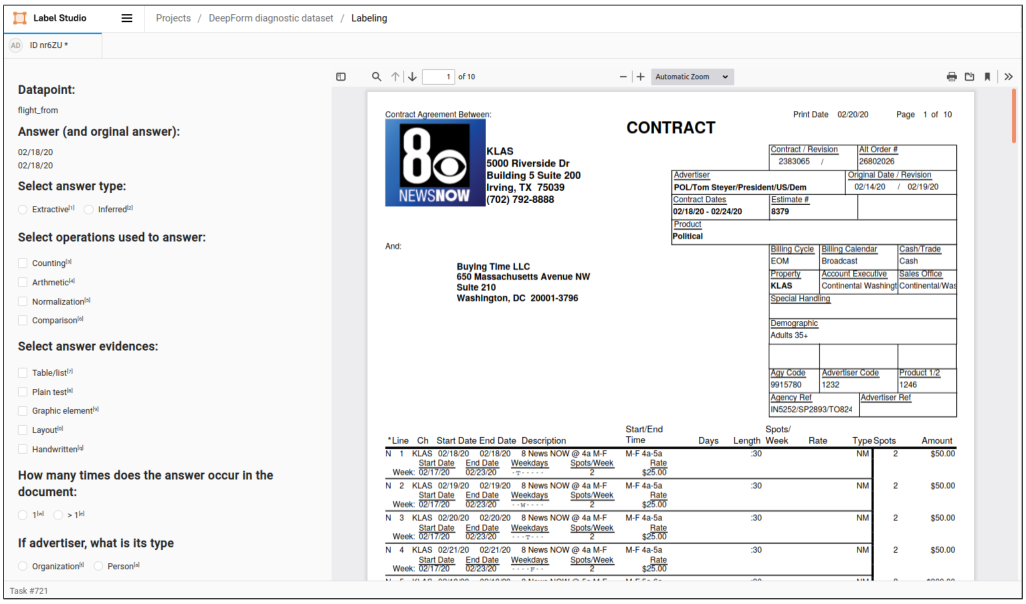


Figure 8: Number of words, pages, and words per page in particular datasets (log scale). Part of the datasets consist only of one-pagers.



Label Studio | Projects / DeepForm diagnostic dataset / Labeling

AD ID nr6ZU *

Datapoint:

flight_from

Answer (and orginal answer):

02/18/20
02/18/20

Select answer type:

○ Extractive[1] ○ Inferred[2]

Select operations used to answer:

□ Counting[3]
□ Arthmetic[4]
□ Normalization[5]
□ Comparison[6]

Select answer evidences:

□ Table/list[7]
□ Plain test[8]
□ Graphic element[9]
□ Layout[0]
□ Handwritten[q]

How many times does the answer occur in the document:

○ 1[w] ○ > 1[e]

If advertiser, what is its type

○ Organization[t] ○ Person[a]

Task #721

1 of 10 Automatic Zoom

Contract Agreement Between:

CONTRACT

Print Date 02/20/20 Page 1 of 10

KLAS
5000 Riverside Dr
Building 5 Suite 200
Irving, TX 75039
(702) 792-8888

And:

Buying Time LLC
650 Massachusetts Avenue NW
Suite 210
Washington, DC 20001-3796

Contract / Revision 2383065 /	Alt Order # 26802026
Advertiser **POL/Tom Steyer/President/US/Dem**	Original Date / Revision 02/14/20 / 02/19/20
Contract Dates **02/18/20 - 02/24/20**	Estimate # **8379**
Product **Political**	

Billing Cycle EOM	Billing Calendar Broadcast	Cash/Trade Cash
Property **KLAS**	Account Executive Continental Washingt	Sales Office Continental/Was
Special Handling		
Demographic Adults 35+		
Agy Code 9915780	Advertiser Code 1232	Product 1/2 1246
Agency Ref IN5252/SP2893/TO824	Advertiser Ref	

*Line	Ch	Start Date	End Date	Description	Start/End Time	Days	Length	Spots/ Week	Rate	Type	Spots	Amount
N 1	KLAS	02/18/20	02/18/20	8 News NOW @ 4a M-F	M-F 4a-5a		:30			NM	2	$50.00
Week:		Start Date 02/17/20	End Date 02/23/20	Weekdays -T----- Spots/Week 2	Rate $25.00							
N 2	KLAS	02/19/20	02/19/20	8 News NOW @ 4a M-F	M-F 4a-5a		:30			NM	2	$50.00
Week:		Start Date 02/17/20	End Date 02/23/20	Weekdays --W---- Spots/Week 2	Rate $25.00							
N 3	KLAS	02/20/20	02/20/20	8 News NOW @ 4a M-F	M-F 4a-5a		:30			NM	2	$50.00
Week:		Start Date 02/17/20	End Date 02/23/20	Weekdays ---T--- Spots/Week 2	Rate $25.00							
N 4	KLAS	02/21/20	02/21/20	8 News NOW @ 4a M-F	M-F 4a-5a		:30			NM	2	$50.00
Week:		Start Date 02/17/20	End Date 02/23/20	Weekdays ----F-- Spots/Week 2	Rate $25.00							

Figure 9: An example of an interface for annotating diagnostic subsets based on document from DeepForm dataset.

It should include as many aspects as possible that are relevant from the perspective of document understanding problem.

Initially, we adopted the taxonomies proposed in DocVQA, Infographics, and TabFact as potential categories [30, 29, 6]. In the next step, we adjusted our taxonomy to all datasets following the previously adopted assumptions, distinguishing seven main categories with 25 subcategories (for a more detailed description of the category (see the section G.1). Then, for each dataset, we prepared an annotation task in the LabelStudio tool [6] (see example 9) along with an annotation instruction. Finally, to determine Human performance, the annotation was carried out by a team of specialists from Applica.ai, where the selected example was noted only by one person.

G.1 Taxonomy description

The taxonomy is based on multiple aspects of documents, inputs, and answers and was designed to be sufficiently generic for future adaptation to other tasks. Here, in each category, we describe the predicates that annotators followed when classified an example into specific subcategories.

Answer source. This category is based on the relation between answer and text in the document.

- Extractive – after lowercasing and white-characters removing, the answer can be exact-matched in the document.
- Inferred – other non-extractive cases.

Output format This category is based on the shape of an output.

- Single value – the answer consists of only one item.
- List – multiple outputs are to be provided.

Output type. This category is based on the semantic of an output.

- Organization – the answer is a name of an organization or institution.

[6] https://labelstud.io/

- Location – the answer is a geographic location globally (e.g., a country, continent, city) or locally (building or street, among others).
- Person – the answer is a personal identifier(name, surname, pseudonym) or its composition. It can have a title prefix or suffix (e.g., Mrs., Mr., Ph.D.) or have a shortened or informal version.
- Number – numerical values given with the unit or percent. Values written in the free text do not comply with this class's definition.
- Date/Time/Duration – the answer represents the date, time, or the difference between two dates or times.
- Yes/No – the answer is a textual output of binary classification, such as Yes/No pairs, and Positive/Negative, 0/1 among others.

Evidence. This category is based on the source of information that allows the correct answer to be generated. When there are multiple justifications based on different pieces of evidence (for example, the address is in a table and block text), it is required to select all the pieces of evidence.

- Table or List – a *table* is a fragment of the document organized into columns and rows. The distinguishing feature of the table is consistency within rows and columns (usually the same data type). Moreover, it may have a header. In that sense, the form is not a table (or at least it does not have to be). A *list* is a table degenerated into one column or row containing a header.
- Plain text – the answer is based on plain text if there is an immediate need to understand a longer fragment of the text while answering.
- Graphic element – the answer is based on graphic evidence when understanding graphically rich, non-text fragments of documents (e.g., graphics, photos, logos (non-text)) are necessary for generating a correct answer.
- Layout – it is evidence when comprehending the placement of text on the page (e.g., titles, headers, footers, forms) is needed to generate the correct answer. This type does not include tables.
- Handwritten – when the text written by hand is crucial for an answer.

Operation. This category is based on the type of operations that are to be performed on the document before reaching to the correct answer.

- Counting – when there is a need to count the occurrences or determine the position on the list.
- Arithmetic – when there is an arithmetic operation applied before answering, or a sequence of arithmetic operations (e.g., averaging).
- Comparison – a comparison in the sense of lesser/greater. Other procedures that a comparison operation can express (e.g., approximation) may be chosen. Here, the operation "is equal" is not a comparison since it is sufficient to match sequences without a semantic understanding.
- Normalization – when we are to return something in the document but in a different form. It may only apply to the output; we do not acknowledge this operation when it is required to normalize a question fragment to match it in the document.

Answer number. This category is based on the number of occurrences of an answer in the document.

- 1 – when there is one path of logical reasoning to find the correct answer in the document. We treat it as one justification for two different reasoning paths based on the same data from the document.
- > 1 – the other cases.

H Unified format

We propose a unified format for storing information in the Document Understanding domain and deliver converted datasets as part of the released benchmark. It assumes three interconnected dataset, document annotation and document content levels. Please refer to the repository for examples and formal specifications of the schemes.

Dataset. The dataset level is intended for storing the general metadata, e.g., name, version, license, and source. Here, the JSON-LD format based on the well-known schema.org web standard is used.[7]

Document. The documents annotation level is intended to store annotations available for individual documents within datasets and related metadata (e.g., external identifiers). Our format, valid for all of the Document Understanding tasks, is specified using the JSON-Schema standard. This ensures that every record is well-documented and makes automatic validation possible. Additionally, to make the processing of large datasets efficient, we provide JSON Lines file for each split, thus it is possible to read one record at a time.

Content. As part of the original annotation or additional data we provide is related to document content (e.g., the output of a particular OCR engine), we introduce the document's content level. Similarly to the document level, we propose an adequate JSON Schema and provide the JSON Lines files in addition. PDF files with the source document accompany dataset -, document-, and content-level annotations. If the source PDF was not available, a lossless conversion was performed.

I Evaluation protocol

Evaluation protocol. All the benchmark submissions are expected to conform to the following rules to guarantee fair comparison, reproducibility, and transparency:

- All results should be automatically obtainable starting from either raw PDF documents or the JSON files we provide. In particular, it is not permitted to rely on the potentially available source file that our PDFs were generated from or in-house manual annotation.
- Despite the fact that we provide an output of various OCR mechanisms wherever applicable, it is allowed to use software from outside the list. In such cases, participants are highly encouraged to donate OCR results to the community, and we declare to host them along with other variants. It is expected to provide detailed information on used software and its version.
- Any dataset can be used for unsupervised pretraining. The use of supervised pretraining is limited to datasets where there is no risk of information leakage, e.g., one cannot train models on datasets constructed from Wikipedia tables unless it is guaranteed that the same data does not appear in WikiTableQuestions and TabFact.
- It is encouraged to use datasets already publicly available or to release data used for pretraining.
- Training performed on a development set is not allowed. We assume participants select the model to submit using training loss or validation score. We do not release test sets and keep them secret by introducing a daily limit of evaluations performed on the benchmark's website.
- Although we allow submissions limited to one category, e.g., QA or KIE, complete evaluations of models that are able to comprehend all of the tasks with one architecture are highly encouraged.
- Since different random initialization or data order can result in considerably higher scores, we require the bulk submission of at least three results with different random seeds.
- Every submission is required to have an accompanying description. It is recommended to include the link to the source code.

J Experiments - training details

The experiments were carried out in an environment with NVIDIA A100-40Gb cards, PyTorch version 1.8.1, and huggingface-transformers in version 4.2.2.

The parameters were selected through empirical experiments with T5-Base model on DocVQA and InfographicsVQA collections. The T5-Large model was used as the basis for finetuning.

The training lasted up to 30 epochs at batch 64 in training, the default optimizer AdamW (lr = 0.0002), and warmup set to 100 updates. Validation was performed five times per epoch, and when

[7] See `https://json-ld.org/` for information on the JSON-LD standard, and `https://developers.google.com/search/docs/data-types/dataset` for the description of adapted schema.

no improvement was seen for 20 validation steps (4 epochs), the training was stopped. The length of the input documents has been truncated to 1024 tokens and the responses to 256 tokens. Dropout was set to 0.15, gradient clipping to 1.0, and weight decay to 1e-05.

The complete source code is attached as the supplementary material.

K Benchmark datasheet

Following Gebru et al. [13] we fill the datasheet for the proposed benchmark. As it was originally designed for datasets, part of the questions might not apply and were skipped.

K.1 Motivation for datasheet creation

Why was the benchmark created? Despite its importance for digital transformation, the problem of measuring how well available models obtain information from a wide range of document types and how suitable they are for freeing workers from paperwork through process automation is not yet addressed. We intend to bridge this major gap by introducing the first Document Understanding benchmark.

Has the benchmark been used already? If so, where are the results so others can compare (e.g., links to published papers)? No, the paper describes the first version of the benchmark.

Who funded the creation dataset? Applica.ai

K.2 Benchmark composition

What are the instances?(that is, examples; e.g., documents, images, people, countries) Are there multiple types of instances? (e.g., movies, users, ratings; people, interactions between them; nodes, edges) Single instance is a PDF document such as report, scientific publication, form, infographic or table excerpted from websites. For each instance in train and dev split we provide associated question-answer or property-value pairs.

How many instances are there in total (of each type, if appropriate)? DocVQA totals $12.8k$ examples, InfographicsVQA totals $5.5k$, Kleister Charity totals $2.7k$, PWC totals $0.4k$m DeepForm totals $1.1k$ WikiTableQuestins totals $2.1k$ and TabFact totals $16.6k$

What data does each instance consist of ? "Raw" data (e.g., unprocessed text or images)? Features/attributes? Is there a label/target associated with instances? If the instances related to people, are subpopulations identified (e.g., by age, gender, etc.) and what is their distribution? OCR layer from scanned PDF, textual question (or property) and textual answer (value), meta-data and diagnostic information.

Is there a label or target associated with each instance? If so, please provide a description. Yes, there is an answer or multiple allowed answers specified for each instance.

Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text. For each instance we provide output form OCR tools (Tesseract, Microsoft Computer Vision API, djvu). However, few documents are problematic for OCR engines and for them we were not able to generate text and layout layer.

Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit. Yes, they contain metadata that informs about the id of the document that was used for the instance. Different instances may share the same underlying document.

Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them. For five out of seven tasks from our benchmark we used original datasets splits. The two datasets in which we changed splits are:

DeepForm. The original DeepForm dataset consists of 2012, 2014, and 2020 subsets differing in terms of annotation quality and documents' diversity. We decided to use only the 2020 subset as for 2014, and 2020 annotations were prepared either automatically or by volunteers, leading to questionable quality. The selected subset was randomly divided into training, development and test set. In addition to the improved 2020 subset, we manually annotated one hundred 2012 documents, as they can pose different challenges (contain different document templates, handwriting, have lower image quality). They were used to extend development and test set. The final dataset consists of 700 training, 100 development, and 300 test set documents.

WikiTableQuestions. The original WTQ dataset consists of training, pristine-seen-tables, and pristine-unseen-tables subsets. We treat pristine-unseen-tables as a test set and create new training and development sets by rearranging data from training and pristine-seen-tables. The latter operation is dictated by the leakage of documents in the original formulation, i.e., we consider it undesirable for a document to appear in different splits, even if the question differs. The resulting dataset consists of approximately 2100 documents divided in the proportion of 65%, 15%, 20% into training, development, and test sets.

Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description. In our benchmark we have two sources of errors:

Annotations. For each task we provided human performance estimation which shows how often the annotators were in agreement with each other (what is the level of annotation noise).

OCR output. As an input for all tasks we used PDF files. Therefore, we used OCR tools (which is no perfect) to retrieve text and layout layer (token bounding boxes).

Is the benchmark self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a future user? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate. Despite the fact that the benchmark aggregates dataset published in various sources it is self-contained. To eliminate some of the barriers in future experiments, we proposed a format to unify varied Document Understanding tasks and convert all of the datasets included in the benchmark. Additionally, we provide versioned OCR layers for scanned documents to make models evaluated in the future directly comparable.

All of these resources are provided on the benchmark website, without a need to download them from external sources.

K.3 Collection Process

What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)? How were these mechanisms or procedures validated? For WikiTableQuestions, we prepare input documents by rendering table-related HTML distributed by authors in *wkhtmltopdf* and crop the resulting files with *pdfcrop*. As these code excerpts do not contain *head* tag with JavaScript and stylesheet references, we use the header from the present version of the Wikipedia website.

As the authors of TabFact distribute only CSV files, we resorted to HTML from the WikiTables dump their CSV were presumably generated from.[8] As Chen et al. [6] dropped some of the columns present in used WikiTable tables, we remove them too, to ensure compatibility with the original TabFact. Rendered files are used analogously to the case of WTQ.

[8] `http://websail-fe.cs.northwestern.edu/TabEL/tables.json.gz`

The remaining datasets had their data kept in the original form. Used procedures were designed and validated in an iterative manner by: (1) validating all generated documents against original source (CSV) and (2) checking a random sample of 200 documents manually looking for anomalies. If any errors were detected, the processing software was fixed and the validation procedure started again.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)? Estimation of human performance for PWC, WikiTableQuestions, DeepForm and annotation of diagnostic subsets was performed in-house (at Applica.ai) by professional annotators in their work time.

K.4 Data Preprocessing

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remainder of the questions in this section. We provide OCR layers for PDF documents to make models evaluated in the future directly comparable.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data. The original PDF files are hosted on the https://duebenchmark.com/data and can be downloaded from there.

Is the software used to preprocess/clean/label the instances available? If so, please provide a link or other access point. To preprocess all documents we have used two OCR tools:

1. Tesseract in version 4.1.1[9]
2. Microsoft Azure Computer Vision API (Azure CV) in version 3.0.0[10]

To estimate human performance and for annotation diagnostic datasets we used open source Label-Studio[11] software (screenshots is provided in the paper appendix for reference).

K.5 Dataset Distribution

How will the dataset be distributed? (e.g., tarball on website, API, GitHub; does the data have a DOI and is it archived redundantly?) All datasets from our benchmark are available on the https://duebenchmark.com/data and can be downloaded from there. Moreover, for each dataset we provide JSON-LD file[12] with detailed description.

When will the dataset be released/first distributed? What license (if any) is it distributed under? We released all datasets already. We used original license for all datasets that we selected to our benchmark.

Are there any fees or access/export restrictions? No.

K.6 Dataset Maintenance

Who is supporting/hosting/maintaining the dataset? Applica.ai

Will the dataset be updated? If so, how often and by whom? No.

If the dataset becomes obsolete how will this be communicated? We will notify users on benchmark site: https://duebenchmark.com/

[9] https://github.com/tesseract-ocr/tesseract/releases/tag/4.1.1

[10] https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/overview-ocr

[11] https://labelstud.io/

[12] https://developers.google.com/search/docs/data-types/dataset

Is there a repository to link to any/all papers/systems that use this dataset? Everyone who want to use our benchmark should submit their results via site https://duebenchmark.com/. The submission should also contain reference to the paper.

Any other comments? We are not planning to update prepared datasets in our benchmark but we consider to prepare second version of our benchmark in the future (with updated list of datasets).

K.7 Legal and Ethical Considerations

Were any ethical review processes conducted (e.g., by an institutional review board)? If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation. In our benchmark we are using datasets which were collected by other researchers and therefore we do not conduct any ethical review processes. Moreover, all datasets are already available.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctorpatient confidentiality, data that includes the content of individuals non-public communications)? If so, please provide a description. No.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why No.

Does the dataset relate to people? If not, you may skip the remaining questions in this section.
DocVQA. No.
InfographicsVQA. No.
Kleister Charity. No.
PWC. Yes.
DeepForm. Yes.
WikiTableQuestions. Yes.
TabFact. Yes.

Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset. No.

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? If so, please describe how.
DocVQA. No.
InfographicsVQA. No.
Kleister Charity. No.
PWC. Yes — we can check what are the authors of the publications.
DeepForm. Yes — in this dataset we are processing receipts from political campaign ads bought around US elections. Sometimes on these forms we could find politician person names.
WikiTableQuestions. Yes - data comes from Wikipedia so we can check person indirectly by going to Wikipedia page from which table was extracted.
TabFact. Yes — data comes from Wikipedia so we can check person indirectly by going to Wikipedia page from which table was extracted.

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? If so, please provide a description. *DocVQA.* No.
InfographicsVQA. No.
Kleister Charity. No.
PWC. No.

DeepForm. Yes — we have information on how much money a given person donated to support the presidency campaign (but this information is publicly available).
WikiTableQuestions. No.
TabFact. No.

Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)? We used data collected by other researchers.

Appendix D: LAMBERT: Layout-Aware Language Modeling for Information Extraction

LAMBERT: Layout-Aware Language Modeling for Information Extraction

Łukasz Garncarek[1(✉)], Rafał Powalski[1], Tomasz Stanisławek[1,2], Bartosz Topolski[1], Piotr Halama[1], Michał Turski[1,3], and Filip Graliński[1,3]

[1] Applica.ai, Zajęcza 15, 00-351 Warsaw, Poland
{lukasz.garncarek,rafal.powalski,tomasz.stanislawek, bartosz.topolski,piotr.halama,michal.turski,filip.gralinski}@applica.ai
[2] Warsaw University of Technology, Koszykowa 75, 00-662 Warsaw, Poland
[3] Adam Mickiewicz University, 1 Wieniawskiego, 61-712 Poznań, Poland

Abstract. We introduce a simple new approach to the problem of understanding documents where non-trivial layout influences the local semantics. To this end, we modify the Transformer encoder architecture in a way that allows it to use layout features obtained from an OCR system, without the need to re-learn language semantics from scratch. We only augment the input of the model with the coordinates of token bounding boxes, avoiding, in this way, the use of raw images. This leads to a layout-aware language model which can then be fine-tuned on downstream tasks.

The model is evaluated on an end-to-end information extraction task using four publicly available datasets: Kleister NDA, Kleister Charity, SROIE and CORD. We show that our model achieves superior performance on datasets consisting of visually rich documents, while also outperforming the baseline RoBERTa on documents with flat layout (NDA F_1 increase from 78.50 to 80.42). Our solution ranked first on the public leaderboard for the Key Information Extraction from the SROIE dataset, improving the SOTA F_1-score from 97.81 to 98.17.

Keywords: Language model · Layout · Key information extraction · Transformer · Visually rich document · Document understanding

1 Introduction

The sequential structure of text leads to it being treated as a sequence of tokens, characters, or more recently, subword units. In many problems related to Natural Language Processing (NLP), this linear perspective was enough to enable significant breakthroughs, such as the introduction of the neural Transformer architecture [28]. In this setting, the task of computing token embeddings is

Ł. Garncarek, R. Powalski, T. Stanisławek and B. Topolski—Equally contributed to the paper.

J. Lladós et al. (Eds.): ICDAR 2021, LNCS 12821, pp. 532–547, 2021.
https://doi.org/10.1007/978-3-030-86549-8_34

solved by Transformer encoders, such as BERT [6] and its derivatives, achieving top scores on the GLUE benchmark [29].

They all deal with problems arising in texts defined as sequences of words. However, in many cases there is a structure more intricate than just a linear ordering of tokens. Take, for instance, printed or richly-formatted documents, where the relative positions of tokens contained in tables, spacing between paragraphs, or different styles of headers, all carry useful information. After all, the goal of endowing texts with layout and formatting is to improve readability.

In this article we present one of the first attempts to enrich the state-of-the-art methods of NLP with layout understanding mechanisms, contemporaneous with [32], to which we compare our model. Our approach injects the layout information into a pretrained instance of RoBERTa. We fine-tune the augmented model on a dataset consisting of documents with non-trivial layout.

We evaluate our model on the end-to-end information extraction task, where the training set consists of documents and the target values of the properties to be extracted, without any additional annotations specifying the locations where the information on these properties can be found in the documents. We compare the results with a baseline RoBERTa model, which relies on the sequential order of tokens obtained from the OCR alone (and does not use the layout features), and with the solution of [31,32]. LAMBERT achieves superior performance on visually rich documents, without sacrificing results on more linear texts.

1.1 Related Work

There are two main lines of research into understanding documents with non-trivial layout. The first one is Document Layout Analysis (DLA), the goal of which is to identify contiguous blocks of text and other non-textual objects on the page and determine their function and order in the document. The obtained segmentation can be combined with the textual information contained in the detected blocks. This kind of method has recently been employed in [17].

Many services employ DLA functionality for OCR (which requires document segmentation), table detection or form field detection, and their capabilities are still expanding. The most notable examples are Amazon Textract [1], the Google Cloud Document Understanding AI platform [8], and Microsoft Cognitive Services [20]. However, each has limitations, such as the need to create rules for extracting information from the tables recognized by the system, or use training datasets with annotated document segments. More recent works on information extraction using DLA include, among others, [2,3,10,14,19,22,25]. They concentrate on specific types of documents, such as invoices or forms, where the layout plays a relatively greater role: more general documents may contain tables, but they can also have large amounts of unstructured text.

The second idea is to directly combine the methods of Computer Vision and NLP. This could be done, for instance, by representing a text-filled page as a multi-channel image, with channels corresponding to the features encoding the semantics of the underlying text, and, subsequently, using convolutional networks. This method was used, among others, by Chargrid and BERTgrid models [5,15]. On the other hand, LayoutLM [32] and TRIE [34] used the image recognition features of the page image itself. A more complex approach was taken by PICK [33], which separately processes the text and images of blocks identified in the document. In this way it computes the vertex embeddings of the block graph, which is then processed with a graph neural network.

Our idea is also related to the one used in [24], though in a different setting. They considered texts accompanied by audio-visual signal injected into a pretrained BERT instance, by combining it with the input embeddings.

LAMBERT has a different approach. It uses neither the raw document image, nor the block structure that has to be somehow inferred. It relies on the tokens and their bounding boxes alone, both of which are easily obtainable from any reasonable OCR system.

1.2 Contribution

Our main contribution is the introduction of a *Layout-Aware Language Model*, a general-purpose language model that views text not simply as a sequence of words, but as a collection of tokens on a two-dimensional page. As such it is able to process plain text documents, but also tables, headers, forms and various other visual elements. The implementation of the model is available at https://github.com/applicaai/lambert.

A key feature of this solution is that it retains the crucial trait of language models: the ability to learn in an unsupervised setting. This allows the exploitation of abundantly available unannotated public documents, and a transfer of the learned representations to downstream tasks. Another advantage is the simplicity of this approach, which requires only an augmentation of the input with token bounding boxes. In particular, no images are needed. This eliminates an important performance factor in industrial systems, where large volumes of documents have to be sent over a network between distributed processing services.

Another contribution of the paper is an extensive ablation study of the impact of augmenting RoBERTa with various types of additional positional embeddings on model performance on the SROIE [12], CORD [21], Kleister NDA and Kleister Charity datasets [27].

Finally, we created a new dataset for the unsupervised training of layout-aware language models. We will share a 200k document subset, amounting to 2M visually rich pages, accompanied by a dual classification of documents: business/legal documents with complex structure; and others. Due to IIT-CDIP Test Collection dataset [16] accessibility problems[1], this would constitute the largest widely available dataset for training layout-aware language models. It would allow researchers to compare the performance of their solutions not only on the same test sets, but also with the same training set. The dataset is published at https://github.com/applicaai/lambert, together with a more detailed description that is too long for this paper.

[1] The link https://ir.nist.gov/cdip/ seems to be dead (access on Feb 17, 2021).

2 Proposed Method

We inject the layout information into the model in two ways. Firstly, we modify the input embeddings of the original RoBERTa model by adding the layout term. We also experiment with completely removing the sequential embedding term. Secondly, we apply relative attention bias, used [11,23,26] in the context of sequential position. The final architecture is depicted in Fig. 1.

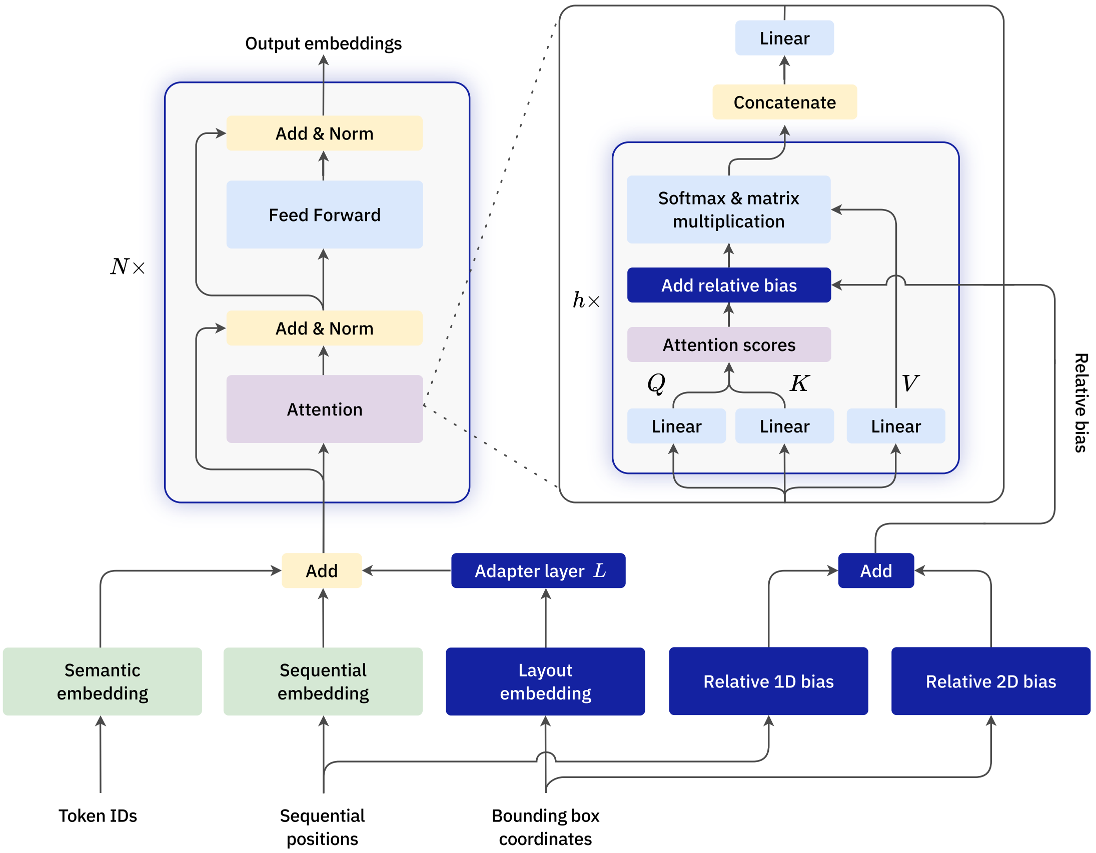


Fig. 1. LAMBERT model architecture. Differences with the plain RoBERTa model are indicated by white text on dark blue background. $N = 12$ is the number of transformer encoder layers, and $h = 12$ is the number of attention heads in each encoder layer. Q, K, and V are, respectively, the queries, keys and values obtained by projecting the self-attention inputs. (Color figure online)

2.1 Background

The basic Transformer encoder, used in, for instance, BERT [6] and RoBERTa [18], is a sequence-to-sequence model transforming a sequence of input embeddings $x_i \in \mathbb{R}^n$ into a sequence of output embeddings $y_i \in \mathbb{R}^m$ of the same length, for the input/output dimensions n and m. One of the main distinctive features of this architecture is that it discards the order of its input vectors. This allows parallelization levels unattainable for recurrent neural networks.

In such a setting, the information about the order of tokens is preserved not by the structure of the input. Instead, it is explicitly passed to the model, by defining the input embeddings as

$$x_i = s_i + p_i, \tag{1}$$

where $s_i \in \mathbb{R}^n$ is the semantic embedding of the token at position i, taken from a trainable embedding layer, while $p_i \in \mathbb{R}^n$ is a *positional embedding*, depending only on i. In order to avoid confusion, we will, henceforth, use the term *sequential embeddings* instead of *positional embeddings*, as the *positional* might be understood as relating to the 2-dimensional position on the page, which we will deal with separately.

Since in RoBERTa, on which we base our approach, the embeddings p_i are trainable, the number of pretrained embeddings (in this case 512) defines a limit on the length of the input sequence. In general, there are many ways to circumvent this limit, such as using predefined [28] or relative [4] sequential embeddings.

2.2 Modification of Input Embeddings

We replace the input embeddings defined in (1) with

$$x_i = s_i + p_i + L(\ell_i). \tag{2}$$

Here, $\ell_i \in \mathbb{R}^k$ stands for *layout embeddings*, which are described in detail in the next subsection. They carry the information about the position of the i-th token on the page.

The dimension k of the layout embeddings is allowed to differ from the input embedding dimension n, and this difference is dealt with by a trainable linear layer $L\colon \mathbb{R}^k \to \mathbb{R}^n$. However, our main motivation to introduce the adapter layer L was to gently increase the strength of the signal of layout embeddings during training. In this way, we initially avoided presenting the model with inputs that it was not prepared to deal with. Moreover, in theory, in the case of non-trainable layout embeddings, the adapter layer may be able to learn to project ℓ_i onto a subspace of the embedding space that reduces interference with the other terms in (2). For instance, it is possible for the image of the adapter layer to learn to be approximately orthogonal to the sum of the remaining terms. This would minimize any information loss caused by adding multiple vectors. While this was our theoretical motivation, and it would be interesting to investigate in detail how much of it actually holds, such detailed considerations of a single model component exceed the scope of this paper. We included the impact of using the adapter layer in the ablation study.

We initialize the weight matrix of L according to a normal distribution $\mathcal{N}(0, \sigma^2)$, with the standard deviation σ being a hyperparameter. We have to choose σ carefully, so that in the initial phase of training, the $L(\ell_i)$ term does not interfere overly with the already learned representations. We experimentally determined the value $\sigma = 0.02$ to be near-optimal[2].

[2] We tested the values 0.5, 0.1, 0.02, 0.004, and 0.0008.

2.3 Layout Embeddings

In our setting, a document is represented by a sequence of tokens t_i and their bounding boxes b_i. To each element of this sequence, we assign its layout embedding ℓ_i, carrying the information about the position of the token with respect to the whole document. This could be performed in various ways. What they all have in common is that the embeddings ℓ_i depend only on the bounding boxes b_i and not on the tokens t_i.

We base our layout embeddings on the method originally used in [7], and then in [28] to define the sequential embeddings. We first normalize the bounding boxes by translating them so that the upper left corner is at $(0, 0)$, and dividing their dimensions by the page height. This causes the page bounding box to become $(0, 0, w, 1)$, where w is the normalized width.

The layout embedding of a token will be defined as the concatenation of four embeddings of the individual coordinates of its bounding box. For an integer d and a vector of scaling factors $\theta \in \mathbb{R}^d$, we define the corresponding embedding of a single coordinate t as

$$\mathrm{emb}_\theta(t) = (\sin(t\theta); \cos(t\theta)) \in \mathbb{R}^{2d}, \tag{3}$$

where the sin and cos are performed element-wise, yielding two vectors in $\mathbb{R}^d$. The resulting concatenation of single bounding box coordinate embeddings is then a vector in $\mathbb{R}^{8d}$.

In [28, Section 3.5], and subsequently in other Transformer-based models with precomputed sequential embeddings, the sequential embeddings were defined by emb_θ with θ being a geometric progression interpolating between 1 and 10^{-4}. Unlike the sequential position, which is a potentially large integer, bounding box coordinates are normalized to the interval $[0, 1]$. Hence, for our layout embeddings we use larger scaling factors (θ_r), namely a geometric sequence of length $n/8$ interpolating between 1 and 500, where n is the dimension of the input embeddings.

2.4 Relative Bias

Let us recall that in a typical Transformer encoder, a single attention head transforms its input vectors into three sequences: queries $q_i \in \mathbb{R}^d$, keys $k_i \in \mathbb{R}^d$, and values $v_i \in \mathbb{R}^d$. The raw attention scores are then computed as $\alpha_{ij} = d^{-1/2} q_i^T k_j$. Afterwards, they are normalized using softmax, and used as weights in linear combinations of value vectors.

The point of relative bias is to modify the computation of the raw attention scores by introducing a bias term: $\alpha'_{ij} = \alpha_{ij} + \beta_{ij}$. In the sequential setting, $\beta_{ij} = W(i-j)$ is a trainable weight, depending on the relative sequential position of tokens i and j. This form of attention bias was introduced in [23], and we will refer to it as *sequential attention bias*.

We introduce a simple and natural extension of this mechanism to the two-dimensional context. In our case, the bias β_{ij} depends on the relative positions of the tokens. More precisely, let $C \gg 1$ be an integer resolution factor (the number of cells in a grid used to discretize the normalized coordinates). If $b_i = (x_1, y_1, x_2, y_2)$ is the normalized bounding box of the i-th token, we first reduce it to a 2-dimensional position $(\xi_i, \eta_i) = (Cx_1, C(y_1 + y_2)/2)$, and then define

$$\beta_{ij} = H(\lfloor \xi_i - \xi_j \rfloor) + V(\lfloor \eta_i - \eta_j \rfloor), \tag{4}$$

where $H(\ell)$ and $V(\ell)$ are trainable weights defined for every integer $\ell \in [-C, C)$. A good value for C should allow for a distinction between consecutive lines and tokens, without unnecessarily affecting performance. For a typical document $C = 100$ is enough, and we fix this in our experiments.

This form of attention bias will be referred to as *2D attention bias*. We suspect that it should help in analyzing, say, tables by allowing the learning of relationships between cells.

3 Experiments

All experiments were performed on 8 NVIDIA Tesla V100 32 GB GPUs. As our pretrained base model we used RoBERTa in its smaller, base variant (125M parameters, 12 layers, 12 attention heads, hidden dimension 768). This was also employed as the baseline, after additional training on the same dataset we used for LAMBERT. The implementation and pretrained weights from the `transformers` library [30] were used.

In the LAMBERT model, we used the layout embeddings of dimension $k = 128$, and initialized the adapter layer L with standard deviation $\sigma = 0.02$, as noted in Sect. 2. For comparison, in our experiments, we also included the published version of the LayoutLM model [32], which is of a similar size.

The models were trained on a masked language modeling objective extended with layout information (with the same settings as the original RoBERTa [18]); and subsequently, on downstream information extraction tasks. In the remainder of the paper, these two stages will be referred to as, respectively, *training* and *fine-tuning*.

Training was performed on a collection of PDFs extracted from *Common Crawl* made up of a variety of documents (we randomly selected up to 10 documents from any single domain). The documents were processed with an OCR system, `Tesseract 4.1.1-rc1-7-gb36c`, to obtain token bounding boxes. The final model was trained on the subset of the corpus consisting of business documents with non-trivial layout, filtered by an SVM binary classifier, totaling to approximately 315k documents (3.12M pages). The SVM model was trained on 700 manually annotated PDF files to distinguish between business (e.g. invoices, forms) and non-business documents (e.g. poems, scientific texts).

In the training phase, we used the Adam optimizer with the weight decay fix from [30]. We employed a learning rate scheduling method similar to the one used in [6], increasing the learning rate linearly from 0 to 1e−4 for the warm-up period of 10% of the training time and then decreasing it linearly to 0. The final model was trained with batch size of 128 sequences (amounting to 64K tokens) for approximately 1000k steps (corresponding to training on 3M pages for 25 epochs). This took about 5 days to complete a single experiment.

After training our models, we fine-tuned and evaluated them independently on multiple downstream end-to-end information extraction tasks. Each evaluation dataset was split into training, validation and test subsets. The models were extended with a simple classification head on top, consisting of a single linear layer, and fine-tuned on the task of classifying entity types of tokens. We employed early stopping based on the F_1-score achieved on the validation part of the dataset. We used the Adam optimizer again, but this time without the learning rate warm-up, as it turned out to have no impact on the results.

The extended model operates as a tagger on the token level, allowing for the classification of separate tokens, while the datasets contain only the values of properties that we are supposed to extract from the documents. Therefore, the further processing of output is required. To this end, we use the pipeline described in [27].

Every contiguous sequence of tokens tagged as a given entity type is treated as a recognized entity and assigned a score equal to the geometric mean of the scores of its constituent tokens. Then, every recognized entity undergoes a normalization procedure specific to its general data type (e.g. date, monetary amount, address, etc.). This is performed using regular expressions: for instance, the date `July, 15th 2013` is converted to `2013-07-15`. Afterwards, duplicates are aggregated by summing their scores, leading to a preference for entities detected multiple times. Finally, the highest-scoring normalized entity is selected as the output of the information extraction system. The predictions obtained this way are compared with target values provided in the dataset using F_1-score as the evaluation metric. See [27] for more details.

4 Results

We evaluated our models on four public datasets containing visually rich documents. The Kleister NDA and Kleister Charity datasets are part of a larger Kleister dataset, recently made public [27] (many examples of documents, and detailed descriptions of extraction tasks can be found therein). The NDA set consists of legal agreements, whose layout variety is limited. It should probably be treated as a plain-text dataset. The Charity dataset on the other hand contains reports of UK charity organizations, which include various tables, diagrams and other graphic elements, interspersed with text passages. All Kleister datasets come with predefined train/dev/test splits, with 254/83/203 documents for NDA and 1729/440/609 for Charity.

The SROIE [12] and CORD [21] datasets are composed of scanned and OCRed receipts. Documents in SROIE are annotated with four target entities to be extracted, while in CORD there are 30 different entities. We use the public 1000 samples from the CORD dataset with the train/dev/test split proposed by the authors of the dataset (respectively, 800/100/100). As for SROIE, it consists of a public training part, and test part with unknown annotations. For the purpose of ablation studies, we further subdivided the public part of SROIE into

Table 1. Comparison of F_1-scores for the considered models. Best results in each column are indicated in bold. In parentheses, the length of training of our models, expressed in non-unique pages, is presented for comparison. For RoBERTa, the first row corresponds to the original pretrained model without any further training, while in the second row the model was trained on our dataset. [a]result obtained from relevant publication; [b]result of a single model, obtained from the SROIE leaderboard [13]

Model	Params	Our experiments				External results	
		NDA	Charity	SROIE*	CORD	SROIE	CORD
RoBERTa [18]	125M	77.91	76.36	94.05	91.57	92.39[b]	–
RoBERTa (16M)	125M	78.50	77.88	94.28	91.98	93.03[b]	–
LayoutLM [32]	113M	77.50	77.20	94.00	93.82	94.38[a]	94.72[a]
	343M	79.14	77.13	96.48	93.62	97.09[b]	94.93[a]
LayoutLMv2 [31]	200M	–	–	–	–	96.25[a]	94.95[a]
	426M	–	–	–	–	97.81[b]	**96.01**[a]
LAMBERT (16M)	125M	80.31	79.94	96.24	93.75	–	–
LAMBERT (75M)	125M	**80.42**	**81.34**	**96.93**	**94.41**	**98.17**[b]	–

training and test subsets (546/80 documents; due to the lack of a validation set in this split, we fine-tuned for 15 epochs instead of employing early stopping). We refer to this split as SROIE*, while the name SROIE is reserved for the original SROIE dataset, where the final evaluation on the test set is performed through the leaderboard [13].

In Table 1, we present the evaluation results achieved on downstream tasks by the trained models. With the exception of the Kleister Charity dataset, where only 5 runs were made, each of the remaining experiments were repeated 20 times, and the mean result was reported. We compare LAMBERT with baseline RoBERTa (trained on our dataset) and the original RoBERTa [18] (without additional training); LayoutLM [32]; and LayoutLMv2 [31]. The LayoutLM model published by its authors was plugged into the same pipeline that we used for LAMBERT and RoBERTa. In the first four columns we present averaged results of our experiments, and for CORD and SROIE we additionally provide the results reported by the authors of LayoutLM, and presented on the leaderboard [13].

Since the LayoutLMv2 model was not publicly available at the time of preparing this article, we could not perform experiments ourselves. As a result some of the results are missing. For CORD, we present the scores given in [31], where the authors did not mention, though, whether they averaged over multiple runs, or used just a single model. A similar situation occurs for LayoutLM; we presented the average results of 20 runs (best run of LAMBERT attained the score of 95.12), which are lower than the scores presented in [31]. The difference could be attributed to using a different end-to-end evaluation pipeline, or averaging (if the results in [31,32] come from a single run).

For the full SROIE dataset, most of the results were retrieved from the public leaderboard [13], and therefore they come from a single model. For the base variants of LayoutLM and LayoutLMv2, the results were unavailable, and we present the scores from the corresponding papers.

In our experiments, the base variant of LAMBERT achieved top scores for all datasets. However, in the case of CORD, the result reported in [31] for the large variant of LayoutLMv2 is superior. If we consider the best scores of LAMBERT (95.12) instead of the average, and the scores of LayoutLM reported in [32], LAMBERT slightly outperforms LayoutLM, while still being inferior to LayoutLMv2. Due to the lack of details on the results of LayoutLM, it is unknown which of these comparisons is valid.

For Kleister datasets, the base variant (and in the case of Charity, also the large variant) of LayoutLM did not outperform the baseline RoBERTa. We suspect that this might be the result of LayoutLM being better attuned to the evaluation pipeline used by its authors, and the fact that it was based on an uncased language model. In the Kleister dataset, meanwhile, performance for entities such as names may depend on casing.

5 Hyperparameters and Ablation Studies

In order to investigate the impact of our modifications to RoBERTa, we performed an extensive study of hyperparameters and the various components of the final model. We investigated the dimension of layout embeddings, the impact of the adapter layer L, the size of training dataset, and finally we performed a detailed ablation study of the embeddings and attention biases we had used to augment the baseline model.

In the studies, every model was fine-tuned and evaluated 20 times on each dataset, except for Kleister Charity dataset, on which we fine-tuned every model 5 times: evaluations took much longer on Kleister Charity. For each model and dataset combination, the mean score was reported, together with the two-sided 95% confidence interval, computed using the corresponding t-value. We considered differences to be significant when the corresponding intervals were disjoint. All the results are presented in Table 2, which is divided into sections corresponding to different studies. The F_1-scores are reported as *increases* with respect to the reported mean baseline score, to improve readability.

5.1 Baseline

As a baseline for the studies we use the publicly available pretrained base variant of the RoBERTa model with 12 layers, 12 attention heads, and hidden dimension 768. We additionally trained this model on our training set, and fine-tuned it on the evaluation datasets in a manner analogous to LAMBERT.

Table 2. Improvements of F_1-score over the baseline for various variants of LAMBERT model. The first row (with grayed background) contains the F_1-scores of the baseline RoBERTa model. The other grayed row corresponds to full LAMBERT. Statistically insignificant improvements over the baseline are grayed. In each of three studies, the best result together with all results insignificantly smaller are in bold. [a]filtered datasets; [b]model with a disabled adapter layer

Train epochs and pages	Embeddings dimension	Inputs				Datasets			
		Sequential	Seq. bias	Layout	2D bias	NDA	Charity	SROIE*	CORD
		●				$78.50_{\pm1.16}$	$77.88_{\pm0.48}$	$94.28_{\pm0.42}$	$91.98_{\pm0.62}$
8×2M	128	●	●			$\mathbf{2.42}_{\pm0.61}$	$0.52_{\pm0.64}$	$0.79_{\pm0.17}$	$0.03_{\pm0.57}$
				●		$1.25_{\pm0.59}$	$\mathbf{2.62}_{\pm0.80}$	$\mathbf{1.86}_{\pm0.15}$	$0.89_{\pm0.83}$
					●	$-0.49_{\pm0.62}$	$2.02_{\pm0.48}$	$0.53_{\pm0.28}$	$-0.17_{\pm0.62}$
				●	●	$0.88_{\pm0.50}$	$\mathbf{3.00}_{\pm0.37}$	$\mathbf{1.94}_{\pm0.16}$	$0.68_{\pm0.62}$
		●		●		$1.74_{\pm0.67}$	$0.06_{\pm0.93}$	$\mathbf{1.94}_{\pm0.18}$	$\mathbf{1.42}_{\pm0.53}$
		●			●	$1.73_{\pm0.60}$	$2.02_{\pm0.53}$	$\mathbf{2.09}_{\pm0.22}$	$\mathbf{1.93}_{\pm0.71}$
		●		●	●	$0.54_{\pm0.85}$	$1.84_{\pm0.42}$	$\mathbf{2.08}_{\pm0.38}$	$\mathbf{2.15}_{\pm0.65}$
		●	●	●		$1.66_{\pm0.76}$	$0.32_{\pm1.35}$	$\mathbf{1.75}_{\pm0.35}$	$1.06_{\pm0.54}$
		●	●		●	$0.85_{\pm0.91}$	$1.84_{\pm0.27}$	$\mathbf{2.01}_{\pm0.24}$	$\mathbf{1.95}_{\pm0.46}$
		●	●	●	●	$\mathbf{1.81}_{\pm0.60}$	$\mathbf{2.06}_{\pm0.69}$	$\mathbf{1.96}_{\pm0.16}$	$\mathbf{1.77}_{\pm0.46}$
8×2M	128	●		●		$1.74_{\pm0.67}$	$0.06_{\pm0.93}$	$\mathbf{1.94}_{\pm0.18}$	$\mathbf{1.42}_{\pm0.53}$
	384	●		●		$0.90_{\pm0.54}$	$0.70_{\pm0.40}$	$\mathbf{1.86}_{\pm0.22}$	$\mathbf{1.51}_{\pm0.60}$
	768	●		●		$0.71_{\pm1.04}$	$0.50_{\pm0.85}$	$\mathbf{2.18}_{\pm0.25}$	$\mathbf{1.54}_{\pm0.51}$
	768[b]	●		●		$0.77_{\pm0.58}$	$\mathbf{2.30}_{\pm0.20}$	$0.37_{\pm0.15}$	$\mathbf{1.58}_{\pm0.52}$
8×2M	128	●	●	●	●	$\mathbf{1.81}_{\pm0.60}$	$2.06_{\pm0.26}$	$1.96_{\pm0.18}$	$1.77_{\pm0.46}$
8×2M[a]		●	●	●	●	$\mathbf{1.86}_{\pm0.66}$	$1.92_{\pm0.19}$	$\mathbf{2.60}_{\pm0.18}$	$1.59_{\pm0.61}$
25×3M[a]		●	●	●	●	$\mathbf{1.92}_{\pm0.50}$	$\mathbf{3.46}_{\pm0.21}$	$\mathbf{2.65}_{\pm0.13}$	$\mathbf{2.43}_{\pm0.19}$

5.2 Embeddings and Biases

In this study we disabled various combinations of input embeddings and attention biases. The models were trained on 2M pages for 8 epochs, with 128-dimensional layout embeddings (if enabled). The resulting models were divided into three groups. The first one contains sequential-only combinations which do not employ the layout information at all, including the baseline. The second group consists of models using only the bounding box coordinates, with no access to sequential token positions. Finally, the models in the third group use both sequential and layout inputs. In this group we did not disable the sequential embeddings. It includes the full LAMBERT model, with all embeddings and attention biases enabled.

Generally, we observe that none of the modifications has led to a significant performance deterioration. Among the models considered, the only one which reported a significant improvement for all four datasets—and at the same time, the best improvement—was the full LAMBERT.

For the Kleister datasets the variance in results was relatively higher than in the case of SROIE* and CORD. This led to wider confidence intervals, and reduced the number of significant outcomes. This is true especially for the Kleister NDA dataset, which is the smallest one. In Kleister NDA, significant improvements were achieved for both sequential-only models, and for full LAMBERT. The differences between these increases were insignificant. It would seem that, for sequential-only models, the sequential attention bias is responsible for the improvement. But after adding the layout inputs, it no longer leads to improvements when unaccompanied by other modifications. Still, achieving better results on sequential-only inputs may be related to the plain text nature of the Kleister NDA dataset.

While other models did not report significant improvement over the baseline, there are still some differences between them to be observed. The model using only 2D attention bias is significantly inferior to most of the others. This seems to agree with the intuition that relative 2D positions are the least suitable way to pass positional information about plain text.

In the case of the Kleister Charity dataset, significant improvements were achieved by all layout-only models, and all models using the 2D attention bias. Best improvement was attained by full LAMBERT, and two layout-only models using the layout embeddings; the 2D attention bias used alone improved the results significantly, but did not reach the top score. The confidence intervals are too wide to offer further conclusions, and many more experiments will be needed to increase the significance of the results.

For the SROIE* dataset, except for two models augmented only with a single attention bias, all improvements proved significant. Moreover, the differences between all the models using layout inputs are insignificant. We may conclude that passing bounding box coordinates in any way, except through 2D attention bias, significantly improves the results. As to the lack of significant improvements for 2D attention bias, we hypothesize that this is due to its relative nature. In all other models the absolute position of tokens is somehow known, either through the layout embeddings, or the sequential position. When a human reads a receipt, the absolute position is one of the main features used to locate the typical positions of entities.

For CORD, which is the more complex of the two receipt datasets, significant improvements were observed only for combined sequential and layout models. In this group, the model using both sequential and layout embeddings, augmented with sequential attention bias, did not yield a significant improvement. There were no significant differences among the remaining models in the group. Contrary to the case of SROIE*, none of the layout-only models achieved significant improvement.

5.3 Layout Embedding Dimension

In this study we evaluated four models, using both sequential and layout embeddings, varying the dimension of the latter. We considered 128-, 384-, and 768-dimensional embeddings. Since this is the same as for the input embeddings of RoBERTa$_{BASE}$, it was possible to remove the adapter layer L, and treat this as another variant, in Table 2 denoted as 768$^{\mathrm{b}}$.

In Kleister NDA, there were no significant differences between any of the evaluated models, and no improvements over the baseline. On the other hand, in Kleister Charity, disabling the adapter layer and using the 768-dimensional layout embeddings led to significantly better performance. These results remain consistent with earlier observations that in Kleister NDA the best results were achieved by sequential-only models, while in the case of Kleister Charity, by layout-only models. It seems that in the case of NDA the performance is influenced mostly by the sequential features, while in the case of Charity, removing the adapter layer increases the strength of the signal of the layout embeddings, carrying the layout features which are the main factor affecting performance.

In SROIE* and CORD all results were comparable, with one exception, namely in SROIE*, the model with the disabled adapter layer did not, unlike the remaining models, perform significantly better than the baseline.

5.4 Training Dataset Size

In this study, following the observations from [9], we considered models trained on 3 different datasets. The first model was trained for 8 epochs on 2M unfiltered (see Sect. 3 for more details of the filtering procedure) pages. In the second model, we used the same training time and dataset size, but this time only filtered pages were used. Finally, the third model was trained for 25 epochs on 3M filtered pages.

It is not surprising that increasing the training time and dataset size, leads to an improvement in results, at least up to a certain point. In the case of Kleister NDA dataset, there were no significant differences in the results. For Kleister Charity, the best result was achieved for the largest training dataset, with 75M filtered pages. This result was also significantly better than the outcomes for the smaller dataset. In the case of SROIE* the two models trained on datasets with filtered documents achieved a significantly higher score than the one trained on unfiltered documents. There was, in fact, no significant difference between these two models. This supports the hypothesis that, in this case, filtering could be the more important factor. Finally, for CORD the situation is similar to Kleister Charity.

6 Conclusions and Further Research

We introduced LAMBERT, a layout-aware language model, producing contextualized token embeddings for tokens contained in formatted documents. The model can be trained in an unsupervised manner. For the end user, the only difference with classical language models is the use of bounding box coordinates as additional input. No images are needed, which makes this solution particularly simple, and easy to include in pipelines that are already based on OCR-ed documents.

The LAMBERT model outperforms the baseline RoBERTa on information extraction from visually rich documents, without sacrificing performance on documents with a flat layout. This can be clearly seen in the results for the Kleister NDA dataset. Its base variant with around 125M parameters is also able to compete with the large variants of LayoutLM (343M parameters) and LayoutLMv2 (426M parameters), with Kleister and SROIE datasets achieving superior results. In particular, LAMBERT$_{BASE}$ achieved first place on the Key Information Extraction from the SROIE dataset leaderboard [13].

The choice of particular LAMBERT components is supported by an ablation study including confidence intervals, and is shown to be statistically significant. Another conclusion from this study is that for visually rich documents the point where no more improvement is attained by increasing the training set has not yet been reached. Thus, LAMBERT's performance can still be improved upon by simply increasing the unsupervised training set. In the future we plan to experiment with increasing the model size, and training datasets.

Further research is needed to ascertain the impact of the adapter layer L on the model performance, as the results of the ablation study were inconclusive. It would also be interesting to understand whether the mechanism through which it affects the results is consistent with the hypotheses formulated in Sect. 2.

Acknowledgments. The authors would like to acknowledge the support the Applica.ai project has received as being co-financed by the European Regional Development Fund (POIR.01.01.01–00–0605/19–00).

References

1. Amazon: Amazon Textract (2019). https://aws.amazon.com/textract/. Accessed 25 Nov 2019
2. Bart, E., Sarkar, P.: Information extraction by finding repeated structure. In: DAS 2010 (2010)
3. Cesarini, F., Francesconi, E., Gori, M., Soda, G.: Analysis and understanding of multi-class invoices. IJDAR **6**, 102–114 (2003)
4. Dai, Z., et al.: Transformer-XL: Attentive language models beyond a fixed-length context. In: ACL (2019)
5. Denk, T.I., Reisswig, C.: BERTgrid: contextualized Embedding for 2D document representation and understanding. In: Workshop on Document Intelligence at NeurIPS 2019 (2019)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (2019)
7. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. In: ICML (2017)

8. Google: Cloud Document Understanding AI (2019). https://cloud.google.com/document-understanding/docs/. Accessed 25 Nov 2019
9. Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., Smith, N.A.: Don't stop pretraining: adapt language models to domains and tasks. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 8342–8360. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.acl-main.740
10. Hamza, H., Belaïd, Y., Belaïd, A., Chaudhuri, B.: An end-to-end administrative document analysis system. In: 2008 The Eighth IAPR International Workshop on Document Analysis Systems, pp. 175–182 (2008)
11. Huang, Y., et al.: Gpipe: Efficient training of giant neural networks using pipeline parallelism. In: NeurIPS (2019)
12. ICDAR: Competition on Scanned Receipts OCR and Information Extraction (2019). https://rrc.cvc.uab.es/?ch=13. Accessed 21 Feb 2021
13. ICDAR: Leaderboard of the Information Extraction Task, Robust Reading Competition (2020). https://rrc.cvc.uab.es/?ch=13&com=evaluation&task=3. Accessed 7 Apr 2020
14. Ishitani, Y.: Model-based information extraction method tolerant of ocr errors for document images. Int. J. Comput. Process. Orient. Lang. **15**, 165–186 (2002)
15. Katti, A.R., et al.: Chargrid: towards understanding 2D documents. In: EMNLP (2018)
16. Lewis, D., Agam, G., Argamon, S., Frieder, O., Grossman, D., Heard, J.: Building a test collection for complex document information processing. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2006)
17. Liu, X., Gao, F., Zhang, Q., Zhao, H.: Graph convolution for multimodal information extraction from visually rich documents. In: NAACL-HLT (2019)
18. Liu, Y., et al.: RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv arXiv:1907.11692 (2019)
19. Medvet, E., Bartoli, A., Davanzo, G.: A probabilistic approach to printed document understanding. IJDAR **14**, 335–347 (2011)
20. Microsoft: Cognitive Services (2019). https://azure.microsoft.com/en-us/services/cognitive-services/. Accessed 25 Nov 2019
21. Park, S., et al.: CORD: A Consolidated Receipt Dataset for Post-OCR Parsing. In: Document Intelligence Workshop at Neural Information Processing Systems (2019)
22. Peanho, C., Stagni, H., Silva, F.: Semantic information extraction from images of complex documents. Appl. Intell. **37**, 543–557 (2012)
23. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(140), 1–67 (2020)
24. Rahman, W., et al.: Integrating multimodal information in large pretrained transformers. In: ACL (2020)
25. Rusinol, M., Benkhelfallah, T., Poulain d'Andecy, V.: Field extraction from administrative documents by incremental structural templates. In: ICDAR (2013)
26. Shaw, P., Uszkoreit, J., Vaswani, A.: Self-attention with relative position representations. In: NAACL-HLT (2018)
27. Stanisławek, T., et al.: Kleister: A novel task for information extraction involving long documents with complex layout (2021) . ArXiv arXiv:2105.05796 Accepted to ICDAR 2021
28. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems 30 (2017)

29. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of ICLR (2019). https://gluebenchmark.com/. Accessed 26 Nov 2019
30. Wolf, T., et al.: Transformers: state-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45. Association for Computational Linguistics, Online (October 2020). https://www.aclweb.org/anthology/2020.emnlp-demos.6
31. Xu, Y., et al.: LayoutLMv2: Multi-modal pre-training for visually-rich document understanding. arXiv arXiv:2012.14740 (2020)
32. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: LayoutLM: pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1192–1200 (2020)
33. Yu, W., Lu, N., Qi, X., Gong, P., Xiao, R.: PICK: Processing key information extraction from documents using improved graph learning-convolutional networks. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 4363–4370 (2021). https://doi.org/10.1109/ICPR48806.2021.9412927
34. Zhang, P., et al.: TRIE: end-to-end text reading and information extraction for document understanding. In: Proceedings of the 28th ACM International Conference on Multimedia (2020)