ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

EVENT EXTRACTION FROM TURKISH TRADE REGISTRY GAZETTE

M.Sc. THESIS

İrem Nur Demirtaş

Department of Computer Engineering

Computer Engineering Programme

16 MAY 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

EVENT EXTRACTION FROM TURKISH TRADE REGISTRY GAZETTE

M.Sc. THESIS

İrem Nur Demirtaş
(504191565)

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Assoc. Prof. Gülşen ERYİĞİT

16 MAY 2023

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TÜRKİYE TİCARET SİCİLİ GAZETESİ'NDEN OLAY ÇIKARIMI

YÜKSEK LİSANS TEZİ

İrem Nur Demirtaş
(504191565)

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Assoc. Prof. Gülşen ERYİĞİT

16 MAYIS 2023

İrem Nur Demirtaş, a M.Sc. student of ITU Graduate School student ID 504191565 successfully defended the thesis entitled "EVENT EXTRACTION FROM TURKISH TRADE REGISTRY GAZETTE", which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Assoc. Prof. Gülşen ERYİĞİT** Istanbul Technical University	.....
Jury Members :	**Assoc. Prof. Ahmet Cüneyd TANTUĞ** Istanbul Technical University	.....
	Assist. Prof. Ayşe Berna ALTINEL GİRGİN Marmara University	.....
		.....

Date of Submission : **14 April 2023**
Date of Defense : **16 May 2023**

To my parents and my brother,

FOREWORD

I would like to thank my advisor for her support and supervision. Over the years we have worked together, not only did I learn a lot, but I also had the chance to develop my curiosity. I was fortunate to work on my thesis as a work project as well. Thus, I would like to thank our head of department Seçil ARSLAN for her unwavering support, both for my thesis and in general. I am happy to have had the opportunity to work with her. I would like to extend my thanks to my managers Alp Güven Buğra AKYÜZ and Güneş AYDINDOĞAN for providing me time and advice whenever I needed. I would like to also acknowledge the help of my teammates, especially during data annotation. Also, I would like to thank my best friend İrem ŞENGÖNÜL for always cheering me on and for our shared experience as long-time schoolmates. Additionally, I thank my parents for always being there for me and for all the love and care they gave. Finally, I would like to thank my brother for being the little brother I actually look up to.

16 May 2023 İrem Nur Demirtaş

TABLE OF CONTENTS

ABBREVIATIONS

TOBB	**:** The Union of Chambers and Commodity Exchanges of Türkiye
TRG	**:** Turkish Trade Registry Gazette
OCR	**:** Optical Character Recognition
EAR	**:** Event Argument Recognition
ERR	**:** Event Relation Recognition
NER	**:** Named Entity Recognition
RE	**:** Relation Extraction
NLP	**:** Natural Language Processing
SOGC	**:** Swiss Official Gazette of Commerce
SEC	**:** US Security and Exchange Commission
EDGAR	**:** Electronic Data Gathering, Analysis, and Retrieval
PDF	**:** Portable Document Format
XML	**:** Extensible Markup Language
NDA	**:** Non-disclosure Agreement
RNN	**:** Recurrent Neural Network
CRF	**:** Conditional Random Field
LSTM	**:** Long Short-Term Memory
BERT	**:** Bidirectional Encoder Representations from Transformers
TRG-EE	**:** Trade Registry Gazette Event Extraction Dataset
TRG-AC	**:** Trade Registry Gazette Announcement Classification Dataset
DC	**:** Document Classification

LIST OF TABLES

Page

LIST OF FIGURES

EVENT EXTRACTION FROM TURKISH TRADE REGISTRY GAZETTE

SUMMARY

The Turkish Trade Registry Gazette is the official gazette published by The Union of Chambers and Commodity Exchanges of Türkiye. Companies announce crucial events like change in management, change in capital or bankruptcy in the gazette. In many industries, the gazette is used as an important source of information and intelligence.

The gazette has a history of almost 70 years. The issues are also publicly available on the internet in image PDF format. This format is both hard to read for humans and hard to process for computers. On top of that, since the gazette has been published in newspaper layout, the text is usually in columns. In later issues of the gazette, some information can be given in tables. Although optical character recognition looks like a viable option for text extraction, it must be supported with image processing.

To extract information from the Turkish Trade Registry Gazette, announcements of selected companies between January 2014 and August 2022 were collected. The collected data consists of PDF documents of gazette pages for the selected companies and related metadata. The metadata contains information about issue number, page number and what type of announcement the company has on the given page. Text was extracted using an image processing and optical character recognition pipeline.

After the text was extracted, it was manually annotated. Since the text is extracted from the whole document, it contains multiple announcements. Thus, announcement boundaries were annotated.

Based on the most important and frequent announcement types encountered in the Turkish Trade Registry Gazette, four event types were defined: Composition with Creditors, Notice to Creditors, Change in Management and Change in Working Capital. Events consist of triggers that signal the occurrence of the event, event arguments that specify general and event-specific entities involved in the events and event roles that define the relations between triggers and arguments. Using these definitions, triggers, arguments and roles were defined and annotated for each of these event types.

Using announcement boundaries, an announcement splitting model was trained. After all collected announcements were split using this model, announcements listed in the metadata table were located in the pages and an announcement classification dataset with 16 announcement types was created. Using this dataset, an announcement classification model was trained. Since announcements are documents of varying lengths, the effect of context was observed. The announcement classification model achieves an F1 score of 0.83.

For trigger and argument extraction, experiments were carried on in different settings. The effect of IOB tags, an added CRF layer and handling argument and trigger extraction separately were observed. The best performing model was determined to be the two-stage one that does not use IOB tags or a CRF layer, with a micro F1 score of 82.5.

For event extraction, a rule-based model and Doc2EDAG [1] were explored. Although the rule-based model performs better on simpler event types, Doc2EDAG was found to be better with a micro F1 score of 73.9 on gold arguments and 54.2 on predicted arguments. Four approaches were proposed to improve the performance. Of these, removing the CRF layer and applying transfer learning yielded improved micro F1 scores of 74.9 and 75.2 over gold arguments and 60.5 and 62.9 over predicted arguments, respectively. The other two proposed methods, namely, turning off path expansion memory and field-aware path expansion yielded poorer results than the baseline.

TÜRKİYE TİCARET SİCİLİ GAZETESİ'NDEN OLAY ÇIKARIMI

ÖZET

Ticaret Sicili Gazetesi Türkiye Odalar ve Borsalar Birliği (TOBB) tarafından 1957 yılından beri yayımlanan resmi bir gazetedir. Türkiye Cumhuriyeti içerisinde hizmet veren ve şartları sağlayan şirketler bir ticaret sicil müdürlüğünde kayıt yaptırırlar. Daha sonra şirketler çeşitli olaylar geçirdikçe bunları müdürlüğe bildirmekle yükümlüdür. Müdürlüklere bildirilen olaylar bir süre sonra Ticaret Sicili Gazetesi'nde yayımlanır. Ticaret Sicili Gazetesi'nde belirlenmiş çeşitli tescil konuları bulunur. Bunlara adres değişikliği, yönetim temsil değişikliği, sermaye artırımı, sermaye azaltımı, konkordato mühleti verilmesi, birleşme, bölünme, çeşitli sebeplerden alacaklılara çağrı gibi konular örnek verilebilir.

1957'den 2022 yılına kadar Ticaret Sicili Gazetesi fiziksel olarak basılmıştır. Aynı zamanda gazetenin tüm sayıları dijitalleştirilmiştir. Eski sayıların taranmış halleri mevcutken, daha yeni sayılar bilgisayarda hazırlandığından, bilgisayarla oluşturulmuş görsel PDF halleri mevcuttur. Gazetenin dağıtımı abonelik yoluyla yapılmaktadır, ancak herkes üyelik oluşturarak belli bir şirketin ilanlarını aratabilir ve görüntüleyebilir. Aramalarda kullanıcıların şirketin bağlı olduğu sicil müdürlüğü bilgisinin yanında sicil numarası veya unvanının en az ilk beş harfini de vermesi beklenmektedir. Abonelik olmadan tam bir gazete sayısına erişmek mümkün değildir. Arama yoluyla erişildiğinde kullanıcılar arattıkları ilanın bulunduğu sayfaların görsel PDF halini indirebilir. Abonelik türüne bağlı olarak kullanıcılar tüm sayıyı görsel PDF veya aranabilir PDF olarak indirebilir.

Ticaret Sicili Gazetesi şirketlerin durumunu anlamak için önemli bir kaynaktır. Örneğin, peş peşe çok şube açan bir şirket için bu durum ileride batacağının göstergesi olabilir. Belli bir yerde yeni şirketlerin açıldığını görmek, onlara hizmet verecek başka şirketlerin yeni müşteriler bulmak için rakiplerine karşı strateji geliştirmesinde yardımcı olabilir. Bir şirketin alacaklılarının şirketin konkordato ilan ettiğinden haberdar edilmesi işlemleri kolaylaştırabilir. Şirketin sermayesini artırması veya azaltması geleceği hakkında iyi veya kötü bir gösterge olabilir. Bu bilgilere erişim bankalar, kargo şirketleri ya da telekomünikasyon şirketleri gibi şirketler için önemlidir.

Ancak günümüzde gazetenin formatı sebebiyle bu bilgiler yapısal bir halde tutulmamaktadır ve bu bilgilere erişim insan gücü gerektirmektedir. Her ne kadar abonelik olmadan bir şirketin ilanları aratılarak takip edilebilse veya abonelikle tam bir sayıya erişilebilse de, gazetenin bir sayısının sayfa sayısının binlere ulaştığı göz önünde bulundurulduğunda bu bilgilerin insanlar tarafından takip edilmesi ve işlenmesinin kolay olmadığı görülmektedir.

Ticaret Sicili Gazetesi'ndeki bilgilere ulaşmak bilgisayarlar için de kolay değildir. Uzun yıllardır gazetenin çoğu sayısının büyük kısmı alışılagelmiş beş sütunlu gazete formatında hazırlanmıştır. 2022 yılında ise bu formattan vazgeçilerek iki sütunlu formata geçilmiştir. Ancak yine de daha az rastlansa da bu iki format dışında başka formatlara da rastlanmaktadır. Üç sütunlu ya da ilanların kutuların içinde yer aldığı formatlara, beş sütunlu formatla başlayıp, kutulu formatla devam eden sayfalara rastlanabilir. İki sütunlu formata 2022 yılından önce de rastlanabilir. Bu formatta bilgilerin paylaşılabileceği daha geniş bir alan olduğundan, bazı bilgiler tablo halinde verilebilir. Örneğin, şirketin kuruluşuna dair, nerede kurulduğu, unvanı, temsilcisinin kim olduğu gibi bilgiler ya da görev değişikliği sonucu yetkilerin nasıl değiştiği gibi bilgiler tablo halinde verilebileceği gibi düz metin içerisinde de ifade edilebilir. Her ne kadar optik karakter tanıma yöntemleri yazıları çıkarabilse de, bu karmaşık yapılar bu sistemleri zorlayabilir. Bu sebeple öncesinde sayfaları uygun yöntemlerle işlemek gerekmektedir.

Ticaret Sicili Gazetesi'nde şirket bazında arama yapılıp ilanların olduğu sayfalara erişilebilse de, bu sayfalarda başka ilanlar da yer almaktadır. İlan içeriklerinin işlenebilmesi için bu ilanların ayrıştırılması gerekmektedir. İlanlar ayrıştırıldıktan sonra sınıflandırılabilir. Türü bilinen ilanlarda şirketlerin geçirdiği olaylar daha detaylı bir seviyede bulunabilir ve bu bilgi yapısal bir hale getirilebilir. İnsanların erişiminin zor ve yavaş olduğu bu bilgiyi yapısal bir halde tutmak daha önce bahsedilen faydaları daha da kıymetli hale getirmektedir.

Yukarıda bahsedilenlerle paralel olarak, bu tezde Ticaret Sicili Gazetesi verisi olay çıkarımı veri kümesi oluşturacak şekilde işlenmiştir. Analiz aşamasında gazetenin çeşitli yıllardaki sayıları incelenmiştir. Çok eski sayılarda tarama sebebiyle gürültülü görseller olduğundan optik karakter tanıma performansı ve ilanların eskiliği göz önünde bulundurularak kapsamdan çıkarılmıştır. Aynı şekilde dijital olarak oluşturulan sayılardaki görüntü kalitesi ve ilan güncelliği düşünülerek Ocak 2014-Ağustos 2021 arasındaki ilanlar kapsama alınmıştır. Öncelikle ilanlar gözle incelenerek 161 şirket kapsama alınmış, etiketleme sırasında 99 şirket daha eklenmiştir.

Kapsama alınan şirketlerin sicil müdürlüğü ve sicil numarası bilgileri TOBB'un portalından sorgulanarak unvanlarıyla birlikte tablo halinde kaydedilmiştir. Daha sonra bu bilgileri doldurup ilanları aratmak için test otomasyonu yöntemiyle tarayıcı üzerinde gerekli alanları otomatik olarak dolduracak bir program hazırlanmıştır. Bu programın yardımıyla şirketlerin belirlenen tarihler arasındaki ilanlarına dair bilgi tablosunu ve ilanlarının bulunduğu gazete sayfalarının görsel PDF formatındaki versiyonları indirilmiştir. Belirlenen yıllar arasında her yıl ortalama 344 ilan olacak şekilde toplam 2751 PDF dokümanı indirilmiştir.

Bu görsel PDF dokümanlarının içinden sayfalar resim halinde çıkarılmıştır. Metni çıkarmadan önce gazete yapısından ve içeride bulunabilecek kutular ve tablolardan kaynaklanabilecek hataları önlemek için bir görüntü işleme aşaması tasarlanmıştır. Metni akışını ve düzenini bozmadan çıkarabilmek için çeşitli boyutlarda kutularda yer alan parçaların ayrıştırılarak işlenmesi gerekmektedir. Bu sebeple öncelikle görüntü işleme yöntemleri kullanılarak kutular tespit edilmiş ve filtrelenmiştir. Tespit edilen kutular ve metnin kalanı ayrı ayrı optik karakter tanıma sistemine verilerek metin çıkarılmıştır. Çıktı piksel koordinatlarını içerdiğinden ve ağaç

yapısında olduğundan bu bilgiler kullanılarak kutuların içeriği metnin geri kalanıyla birleştirilmiştir. Sayfaların genel yapısı ve tablo içerikleri bu sayede korunmuştur.

489 dokümandan çıkarılan metinler yedi etiketleyici tarafından ilan sınırları ve olay çıkarımı için etiketlenmiştir. Etiketlenen ilan sınırları kullanılarak bir ilan ayrıştırma modeli geliştirilmiştir. İlan ayrıştırma modeline bir satır verildiğinde bir BERT modeli yardımıyla kelime gösterimlerini oluşturur ve BERT modellerinde bulunan CLS simgesinin temsilini bir lineer katmana aktararak verilen satırı başlangıç, bitiş veya ara satır olarak sınıflandırır. Bu model 0.94 F1 skoruyla eğitildikten sonra ilanların ayrıştırılması için kullanılmıştır.

İlanlar indirilirken hedeflenen bir ilanın sayfası ve konusu indirildiği için, tüm ilanlar ayrıştırıldıktan sonra sicil numarası bilgisi kullanılarak tabloda yer alan ilan ayrıştırılan ilanın metniyle eşleştirilmiştir. Bu işlem 0.94 doğrulukla gerçekleştirilmiştir. Ayrıştırılan ilan metinleri konularıyla eşleştirildikten sonra konular tekilleştirilmiştir. Ticaret Sicili Gazetesi'nde tekrarlı konular bulunmaktadır. Örneğin, kuruluş olayı anonim şirketler, limited şirketler, iş ortaklığı işletmeleri ve gerçek kişi ticari işletmeleri için ayrı konular olarak geçmektedir. Bu sebeple bu tarz konular tekilleştirilmiştir. Toplanan veride en sık görünen 15 konu ve geriye kalan tüm konular 16 sınıfla temsil edilerek bir ilan sınıflandırma veri kümesi oluşturulmuştur.

İlan türlerine bağlı olarak ilanlar değişik uzunluklarda olabilir ve bazı ilan türleri benzer dilde yazılmış olabilir. Günümüzde sıklıkla kullanılan dil modellerinin kapasitesi doküman işleyecek kadar geniş değildir. Örneğin, bu tezde kullanılan BERT modeli en fazla 512 sembolle çalışabilir. Dokümandaki cümle sayısı arttıkça işleme performansı ve süresi de etkilenmektedir. Bu sebeple bir ilan sınıflandırma modeli eğitilmiş ve ilanların ilk 5, 10 ve 25 cümleyle eğitildiğinde modelin performansı gözlemlenmiştir. Model öncelikle her bir cümleyi BERT dil modeliyle işleyerek gösterimleri oluşturur. Daha sonra tüm kelimeler üzerinde maksimum işlemi uygulayarak gösterimlerini tek bir vektöre indirger ve bu şekilde cümle gösterimini oluşturur. Daha sonra bu gösterimler birleştirilip lineer katman tarafından işlenerek cümleler arası bilgi geçişi sağlanır. Oluşan gösterimler üzerinde bir kez daha maksimum işlemi tekrarlanarak doküman gösterimi elde edilir ve bu gösterime lineer katman yardımıyla 16 sınıftan biri atanır. Sınıflandırma 0.83 F1 skoruyla gerçekleştirilmiştir. Cümle sayısınının özellikle yeterince örnek olduğunda faydalı olduğu görülmesine rağmen, doküman işleme zamanını da lineer olarak artırdığı gözlemlenmiştir.

Olay çıkarımı probleminde bir metinde serbest şekilde yazılmış bir olayın belirlenmiş bir formatta gösterilmesi hedeflenmektedir. Bu hedef doğrultusunda Ticaret Sicili Gazetesi'nde sık görülen ve şirketlerin durumu konusunda ayırt edici olan dört ana olay türü belirlenmiştir. Bunlar Konkordato, Alacaklılara Çağrı, Yönetim Değişikliği ve Sermaye Değişikliği olarak listelenebilir. Bu olayların bulunduğu ilanlar incelenmiş ve her biri için tetikler ve argümanlar belirlenmiştir. Literatürdeki diğer olay çıkarımı veri kümelerinden farklı olarak bu olaylar yardımcı varlıklar da içermektedir. Bu yardımcı varlıklar İnsan, Unvan, Para ve Yetki Türü olarak sıralanabilir. Yardımcı varlıkların her biri bir tetik ve farklı sayılarda argümanlar içerir. Etiketleyiciler tüm dokümanlarda bu olayları işaretlemişlerdir. Olay türüne bağlı olarak bir ilan birden fazla olay içerebilir. Tüm bu olaylar doküman seviyesinde işaretlendiğinden, tetikler

ve argümanlar farklı cümlelerde yer alabilir. Türkçe için doküman seviyesindeki ilk olay çıkarımı veri kümesi bu veri kümesidir. Ticaret Sicili Gazetesi olay çıkarımı veri kümesi 1284 ilan üzerinde, 11818 tetik ve argüman etiketi ile tetiklerle argümanlar arasındaki ilişkileri gösteren 14311 rol etiketi içermektedir.

Tetik kelimesi ve argüman çıkarımı için çeşitli değişkenlerle deneyler yapılmıştır. IOB etiketlerinin, şartlı rastgele alan katmanının ve tetik ve argümanların ayrı ayrı çıkarılmasının etkisi gözlemlenmiştir. En iyi sonuç IOB etiketlerini ve şartlı rastgele alan katmanını kullanmayan model ile elde edilmiştir. Model orijinal argümanlar üzerinde 73.9, tahmin edilen argümanlar üzerinde 54.2 mikro F1 skoruna ulaşmıştır.

Olay çıkarımı için kural tabanlı bir model ile Doc2EDAG [1] modeli kullanılmıştır. Kural tabanlı model daha basit olaylarda daha iyi performans elde ediyor olsa da Doc2EDAG'ın gerisinde kalmıştır. Doc2EDAG orijinal argümanlar ile 73.9, tahmin edilen argümanlar ile 54.2 mikro F1 skoru elde etmiştir. Modelin performansını iyileştirmek için dört yaklaşım önerilmiştir. Bunlardan koşullu rastgele alan katmanını kaldırmak ve öğrenme aktarımı, original argümanlar üzerinde hesaplanan mikro F1 skorunu sırasıyla 74.9 ve 75.2'ye, tahmin edilen argümanlar üzerinde ise sırasıyla 60.5 ve 62.9'a çıkarmıştır. Önerilen diğer iki yöntem olan yol genişletme hafızasını kapatmak ve alan odaklı yol genişletme ise performansı kötüleştirmiştir.

1. INTRODUCTION

This thesis introduces a new Turkish dataset in the finance and trade news domain and provides baselines in related tasks, namely, announcement classification, event argument and trigger extraction and event extraction. The source of this data is the Turkish Trade Registry Gazette (TRG) (Türkiye Ticaret Sicili Gazetesi in Turkish), which is the official gazette of Turkish trade news, published by The Union of Chambers and Commodity Exchanges of Türkiye (TOBB). In this gazette, companies inform TOBB of any events they undergo and these are published for public access by TOBB in the form of gazette issues. The announcements are of importance in many sectors for monitoring and intelligence purposes, such as banking, telecommunication and postal services. Since the gazette is published in image PDF format, a text extraction pipeline involving image processing and OCR is designed and implemented after the data collection phase. Using the extracted text, a document splitting dataset, an document classification dataset and a document-level event extraction dataset that has both event argument and role labels is produced. The dataset is manually labeled and models that perform announcement splitting, announcement classification, event argument and trigger extraction and event extraction are implemented and their performances are reported.

As mentioned earlier, the TRG comes in image PDF format. The layout of the pages are not easy for OCR tools, as they usually contain columns, boxes and tables, like newspaper pages. Moreover, the layout of pages change year by year and even within the same issue, there may be multiple layouts. This necessitates a processing pipeline that handles harder parts for OCR tools and retains document structure to ensure correct order while extracting text.

Although TRG pages contain signals that show announcement boundaries, this information is not machine-readable and cannot be extracted with easy rules. Even after the all the text is extracted from pages, boundaries of individual announcements

must be determined. Since boundaries of documents are not uniform and are also distorted by errors introduced by OCR, a flexible approach must be taken in announcement splitting.

The TRG offers many announcement categories. Announcements in an issue or page do not necessarily have to belong to the same category. Moreover, the announcement type may or may not be declared in the announcement text. In order to extract maximum amount of information from a given page, it is important to classify all announcements according to announcement types. This requires a document classification solution.

To find predefined events in a given announcement, two tasks have to be performed: event argument recognition (EAR) and relation extraction (RE). Events are mainly made up of three elements: Triggers, arguments and roles. Triggers are series of tokens that signal the occurence of an event, arguments are token spans involved in the event and roles are relations between triggers and arguments. These entities can be classical entities found in named entity recognition (NER), like time expressions or person names or event-specific entities. Although entities of similar or the same type can be found around the context, the goal is to find those that actually belong to an event, unlike NER. In addition, the TRG may involve complex time expressions and durations that refer to other entities, censored sensitive information and distortions caused by OCR, making the problem challenging.

As defined in a survey [2], event extraction is a subproblem in information extraction in which the goal is extracting structured event information from unstructured text where the event consists of triggers that signal its occurence or a state change and arguments that parametrize it in a way that answers 5W1H questions. Information extraction can cover other problems like named entity recognition or relation extraction on their own. In event extraction, usually these two tasks are performed one after the other to determine the trigger, the arguments and how they are related.

Also for the TRG, in the event extraction step, the relations between the triggers and arguments of events should be determined. Four types of events are targeted and multiple instances of four of these events can be found in announcements.

In documents that have Composition with Creditors events, only a single event occurs. Furthermore, arguments of an event instance may be scattered across multiple sentences. Thus, document-level event extraction must be conducted. To this end, rule-based and deep learning-based models are applied to provide baselines.

1.1 Literature Review

Information extraction is an important task in NLP and real-life applications. It can be applied in numerous domains ranging from social media texts to legal documents. Information extraction is especially important in finance, since there are rich sources of information, like documents and websites. In many domains, news is an important source of information too. Information extracted from news can be used to keep track of developments in any given field.

Event extraction has been studied in two settings: sentence-level and document-level. Although sentence-level event extraction has been well-studied, it fall short in most settings since events may not be independent from the context of the documents they are in. Event extraction was also covered in many domains such as medicine, social sciences and finance.

In [3], the authors work on sentence-level argument extraction and argue that while triggers imply the existence of an event and events are usually constructed around them, depending on trigger extraction as the first step of event extraction may cause error accumulation later steps. They also highlight that meanings of labels and relations between arguments are not utilized in event extraction. To address this issues they propose a new multi-task framework where train a model on sentence event identification, argument extraction and event detection jointly. After calculation token representation with BERT and retrieving event type embeddings produced by an embedding layer, the authors pass these representations to different sub-models. The sentence event identification model detects the event present in the sentence using the representation of the CLS token. A stack of multi-head attention layers takes token representations and event type embeddings and classifies event types signalled by each word. Using outputs of these models, the argument extraction model produces event triples of event type, argument and role and classifies valid triples. This way the authors

are able to extract events without relying solely on triggers. The multi-task setting helps the model achieve better performance compared to previous models.

Two important issues in event extraction are overlapping trigger and overlapping argument problems as highlighted by [4]. The first problem refers to words that serve as trigger in different event types and the second refers to same type of arguments that appear in different roles in different event types. Another is the existence of multiple events in the same sentence or document. All three of these problems must be addressed in both sentence-level and document-level event extraction.

Another problem that is specific to document-level event extraction is the arguments scattering problem. In this problems arguments of an event scatter across different sentences, requiring document-level understanding for correct extraction. Different datasets and models in literature try to address these problems.

In the domains of finance and news, many datasets were introduced for different information extraction tasks.

In [5], the authors introduced two new datasets for key information extraction: Kleister-NDA and Kleister-Charity. For both datasets, the goal is extracting document-specific information. The data sources for Kleister-NDA and Kleister-Charity are non-disclosure agreement (NDA) and annual financial reports for charities in PDF format, respectively. The authors highlight that these are long documents with complex layouts. Text from the documents were extracted with OCR. Three annotators annotated four and eight types of entities in Kleister-NDA and Kleister-Charity datasets, respectively. The Kleister-NDA dataset consists of 540 documents containing a total of 2160 entities and Kleister-Charity consists of 2788 documents with 21612 entities. Kleister-NDA was manually annotated, but Kleister-Charity was automatically labeled since the necessary information was obtained with the documents from Charity Commission. In Kleister-NDA, 38% of the documents were separated for the test set. Kleister-Charity was split into train, validation and test sets with 65:15:20 ratio. They process the documents in sliding windows of 300 tokens. They perform token classification with RNN + CRF and transformer models and report scores for baseline performance.

In [6], the authors collect an event detection dataset from various websites for 11 different event types. The dataset consists of 2266 documents matched with corresponding event types, such as acquisition, stock split or dividend. Two annotators manually annotated parts of documents that indicate the given event. To construct the training set, documents from each category is sampled in a balanced way and combined with documents that do not contain events. Based on these events, they make buy-sell decisions on the stock market and report performance in terms of win-rate, average return and excess returns. For prediction, they run BERT [7] on the first 256 tokens of the documents. A low-level detector predicts event probabilities at the token level and a high-level detector merges the scores the document level representation constructed using token and "[CLS]" representations. High-level detector's output is used for event prediction.

In [8], the authors introduce a dataset for event extraction from commodity news. They collect text from 8 English news websites and select commodity news based on headlines. The annotators annotate event triggers, arguments, roles and metadata. The authors then apply data augmentation by replacing triggers using FrameNet and arguments based on named entity types. The dataset covers 18 different types of events 21 entity types.

In [9], the authors collect crude oil news from an English website and manually annotate sentence-level events with triggers, arguments and roles and metadata. They apply the same data augmentation techniques in [8] and further expand the dataset using active learning. The final dataset consists of 425 documents, 7059 sentences, 10578 events and 22267 arguments. For argument detection, event properties classification, they employ BERT and perform token classification. For event extraction, they use the Joint Multiple Event Extraction model introduced in [10].

The Joint Multiple Event Extraction model is a sentence-level event extraction model trained on the ACE 2003 corpus [10,11]. ACE 2003 is a sentence-level event extraction dataset constructed from news text in four languages. In [10], the authors process sentences with a bidirectional LSTM network and use a graph convolutional layer network to model relations between tokens. The word representations fed to the

network contain POS tags, entity types and position embeddings as well. The output of the graph convolutional layer is used to predict roles of other detected entities given a trigger and classifier classifies the trigger. The authors report state-of-the-art performance for the task at the time.

Much of the work on event extraction in finance has been carried out in Chinese. Three datasets stand out in event extraction in Chinese. FewFC is a sentence-level event extraction dataset introduced by [12], collected from Chinese news reports and company announcements.

FewFC is a sentence-level event extraction dataset used by [12]. The dataset is collected from Chinese news reports and company announcements. The dataset defines 10 different types of events in the financial domain. The dataset was released as part of a competition. One of the notable works that reports scores on the this dataset is [12]. In this work, the authors handle EAR and ERR jointly by modeling the problem as a dual question answering problem. The model consists of three decoders, two of which encode questions. The other encoder encodes the sentence. Along with the sentence, two questions of the form "What plays <role> in <event>?" and "What is the role of <argument> in <event>?" are asked. The outputs of the questions encoders are merged with that of the sentence encoder with attention and two separate classifiers classify the arguments and roles. In [13], the authors use BERT and BiLSTM to construct sentence and entity representations. Based on shared entities, they represent the document as a graph and integrate vertex features. Using graph attention networks, they predict sentence communities, which represent different events. Based on the predict event type, they use an argument classifier to extract the event. They train the model on FewFC.

In [4], the authors work on FewFC and propose an encoder-decoder model with BERT as encoder and three cascaded decoders that decode event types, boundaries of event triggers and boundaries of event arguments with roles. They utilize attention and conditional fusion layers to merge information from previous steps. They address overlapping trigger and argument problems, but leave error propagation for future work.

In [14], the authors collect a Chinese financial event extraction dataset and label it using distant supervision in two steps: First, sentence-level events are detected using a dictionary for triggers and a knowledge base for arguments. To validate the accuracy of this method, they manually annotate 200 samples. They assume sentences with a specific trigger and the largest number of arguments contain events. Then, at the document level, such sentences are marked as positive samples and the rest is marked negative. They propose a two-stage model in which the first stage extracts arguments and triggers at the sentence-level using BiLSTM-CRF and the second stage produces sentence-level and document-level representations using a CNN, the output of which is corrected for missing arguments based on hand-crafted rules. They compare their performance against a pattern-based baseline and report much better performance.

ChFinAnn is the latest and largest Chinese document-level extraction dataset introduced in [1]. The authors use distant supervision to label events in stock exchange news using a knowledge base. The authors release ChFinAnn and propose a model that eliminates the need for trigger detection by converting the problem to table filling carried out using directed acyclic graphs. This way, event argument extraction and event relation recognition steps are merged in an end-to-end model. They use a transformer model on sentences to detect entities and produce sentence and entity representations by max-pooling over token representations. The document-level information between sentences are integrated into these representations with the help of another transformer. After the sentences and entity representations are created, a third transformer is used for path expansion classification on the directed acyclic graph (DAG) that represents how the table is filled. The nodes on the DAG represents columns of the table. On each node, the transformer uses a memory tensor and an entity tensor to classify entities for that role. This work achieves the state-of-the-art and addresses arguments scattering and multi-event problems, but remains too resource-intensive as pointed out by [15].

In [16], the authors enhance entity representations in Doc2EDAG with graph embeddings they calculate on a knowledge graph using graph neural networks. They also extend ChFinAnn with lawsuit events and report superior performance in comparison to [1]. The authors employ the graph embeddings at the path expansion

stage of the model to model document context better. They use the knowledge graph to label data with distant supervision as well. With the addition of graph embeddings, the authors report superior performance over Doc2EDAG.

In [17], the authors state that the straightforward role-based evaluation in event extraction averages over role scores for events overestimate performance in settings where false positives and false negatives are present. They propose two evaluation metrics, one that considers an event false altogether if false positives are present and another that also does not allow false negatives. Since these metrics are more strict, they result in a lower performance. To aid this problem, they remodel the path-expansion part of Doc2EDAG as a Markov decision process (MDP). In the MDP, already extracted arguments are kept as the state and at each step, a binary action is taken to expand the DAG for the considered node or not. This helps them optimize the model with the proposed metrics and result in better performance.

In [18], inspired by the way humans read things, the authors alter Doc2EDAG to incorporate rough and elaborate reading mechanisms. They focus on producing redundancy-aware representations by removing redundant information in sentence and document representation by subtracting irrelevant information in the produced embeddings. They also introduce an entity copying mechanism to extract entities. This study highlights the importance of memory and removal of redundant information in event extraction.

In [19], the authors criticize [1] and [14] for inefficiency due to their sequential decoding mechanisms and propose a new model with parallel decoding capability. After forming document-aware sentence and entity representation using a transformer, they use two other transformers to decode roles and events. An event-to-role decoder merges the outputs of these two and predicts possible events. Using a Hungarian algorithm-inspired loss function, they find the best bipartite match for the ground truth and predict all events in parallel. The downside of the proposed model is that the number of predicted events is also a hyperparameter to tune.

In [20], the authors employ graph neural networks on event extraction on ChFinAnn. Similar to other work, they extract entity representation on sentences using a

transformer model coupled with a CRF layer. To construct entity and sentence representations, they take the mean of the token representations over their spans. Then, each of these representations are treated as nodes on the graph. Between the nodes, sentence-sentence, sentence-mention, inter-mention and intra-mention edges are introduces to model how they interact with each other. The entity and sentence representations enhanced with graph representations are then used for event type classification and event extraction. The path expansion approach is similar to [1] as well. At each node of the graph, they predict which entity can be used to fill the role. To make the predictions, they use a transformer model with a global memory module called tracker. For every path the model considers, both the completed and the uncompleted paths are kept in the memory. They perform ablation studies to show the effect of different kinds of edges introduced in the graph representation and the different types of information tracked by the tracker module. The absence of any of these causes decrease in the performance. In both single-event and multi-event settings, the proposed model outperforms Doc2EDAG in all comparisons.

In [21], the authors propose new entity-relation modeling and path expansion approaches over Doc2EDAG. They use a BERT model with CRF layer to obtain entity representations, after that for each entity pair in the extracted entity set, they classify the relation between the pairs. Since they also follow the table-filling approach of Doc2EDAG, the relations are defined as field pairs in the table. For all possible relations, entity-entity matrices that represent the relations are constructed. Co-reference and co-existence matrices are also calculated for entities with the same surface form and entities that are found in the same sentence respectively. To utilize these matrices, the authors propose the Relation Augmented Attention Transformer, which attends over the matrices separately and combines the information. The relation augmented attention calculation is done parallel to the self-attention calculation in the transformer and their outputs are combined. They replace the transformer that produces the path representations in [1] and the one that updates path memory with their proposed transformer. This way, the relations between different entities are also considered during path expansion. They compare their model with [1], [22], [14] and [20] on two datasets. Although the proposed model is outperformed by other

Table 1.1 : Comparison of models trained on ChFinAnn. All scores were retrieved from their respective papers.

Model	Year	P	R	F1
Doc2EDAG [1]	2019	81.1	77	79
DE-PPN [19]	2021	N/A	N/A	77.9
Doc2EDAG + Knowledge Graph [16]	2020	84.4	79.1	81.7
SCDEE [13]	2021	N/A	N/A	76.3
GiT [20]	2021	82.3	78.4	80.3
PT-PCG [22]	2022	83.7	75.4	79.4
HRE [18]	2022	81.7	72.5	76.8
RAAT [21]	2022	84.0	79.9	81.9

models in some event types, it provides a better performance overall. The model is especially better than others in multi-event scenarios.

In [22], the authors detect entities at the sentence level with an LSTM and after constructing entity and sentence representations, they represent the document as a graph by constructing an adjacency matrix. Based on their assumption, nodes with an out-degree above a threshold are considered pseudo-triggers and nodes without out-going edges are considered arguments. After pruning the graphs based on these rules, they predict event types and match nodes with roles. The authors compare their model on ChFinAnn and report superior performance against previous methods. The authors also report performance on DuEE-fin [23], which is an event extraction dataset that has triggers unlike ChFinAnn. DuEE-fin is introduced by [23] and is a manually annotated financial event extraction dataset for Chinese. DuEE is a large scale dataset defines 13 event types and 92 argument types. The events are labeled at the document level and arguments can be in complex relations where they play multiple roles in the event or exist as argument in multiple events.

Comparison of models that use ChFinAnn and improve upon Doc2EDAG can be found in Table 1.1

DuEE in [24] is also an event extraction dataset for Chinese that was manually labeled on real-life online news texts. The authors annotated 65 different events and 121 different types of arguments on the sentence level.

In [25], a new dataset for Turkish financial entity recognition was introduced. The authors manually annotate 500 documents collected from finance section of a website for 16 generic and 16 financial entities. They provide a BERT-based baseline and report an F1 score of 0.654 on 31497 entities.

In [26], the authors compare BERT-based models and CRF in event classification and argument classification. The authors use a sentence-level financial event extraction dataset they collected using online stock exchange news. The dataset is initially manually labeled and later expanded via active learning and weak supervision. They observe the effect of active learning and weak supervision on performance and report the positive effect especially weak supervision has on the performance.

In [27], the authors introduce a dataset for document classification, event detection and event extraction for protest events. The data source for the dataset is online news in English for protest events in India and China and South Africa. The dataset is manually annotated by experts in social and political sciences. In the first round of annotation, the annotators classified documents as containing or not containing protest events. Then, they classified sentences in the documents that contain events in the same manner. Then, for the sentences that contain protest events, the annotators annotated event triggers and arguments. The authors also provide baselines for each of the three tasks.

In [28], the authors extract transaction from scanned money transfer documents received by a bank. The document layout changes by customer and the text is extracted using an OCR tool. The authors define a transaction structure that includes sender, receiver and process details divisions. In these divisions, fields regarding the transaction are filled. As the first step, NER is used to extract information regarding the fields. In the second step, binary relation extraction is performed between entities using entity representations produced by an LSTM. After the entities are assigned to divisions, the authors run a maximal clique factorization-based algorithm on the produced graph to decode transactions. The authors report performance on models trained with different language models. They also investigate the effect of positional features in the model, since the document layout is informative.

The comparison of the above mentioned datasets can be found in Table 2.15 and Table 2.16 in Chapter 2.

1.2 Other Related Work

Optical character recognition (OCR) is the problem of extracting characters from a given image. This is a well-studied problem and there exists many tools offered by both cloud providers, open-source developers and by many other companies. Tesseract has been around since 1980s and was made open source later on [29]. Over the years, Tesseract was further improved with language model and LSTM support for improved performance in over 100 languages [30]. Thanks to being open source, Tesseract free of charge. It can also be run locally. Tesseract also has addons that provide developers wrappers for various languages. Tesseract was chosen for OCR phases in this thesis due to open-source availability, Python support for easier integration in pipelines and support of Turkish in its LSTM model.

Transformer is a type of neural network architecture developed by Vaswani et. al. [31]. It is mainly composed of feed-forward, layer normalization and attention layers that help the model to consider relations between different parts of the input. The architecture is made up of encoder and decoder parts. The encoder and the decoder contain encoder and decoder blocks respectively. The encoder block has a multi-head attention layer, followed by layer normalization, a feed forward layer and finally another layer normalization layer. The decoder has a masked multi-head attention layer and another layer normalization layer before the mentioned layers. Positional encoding is added to the input in both parts. Among these layers, the multi-head attention layer is introduced by the authors. Multi-head attention layer takes as input query, key and value vectors. These are processed by a scaled dot-product attention layer that calculates a weight for each query and key pair by performing matrix multiplication, scaling and softmax operations and performs another matrix multiplication with the calculated scores and the value vectors to calculate the output. Multiple scaled dot product attention layers are used in the multihead attention layer, the output of which are concatenated and passed to a linear layer. In the masked variant, a mask is applied after the dot product of queries and keys are calculated.

The authors prove the effectiveness of the model by outperforming the state-of-the-art models at the time in two machine translation tasks. Transformer models have since become a popular architecture not only in NLP, but also in other areas of deep learning. For example, Vision Transformer model in [32] the authors treat patches of images as tokens and apply transformers on image recognition tasks. In [33], the authors employ transformers in encoder-decoder configuration to develop a speech-to-text that can perform English and multilingual language recognition, translation and language identification.

The effectiveness of the transformer architecture led to the development of many transformer-based language models. One of the earlier and most popular of these is BERT [7]. BERT uses the encoder model in [31]. The authors experiment with different number of layers and hidden sizes in the encoder model. In order to represent tokens by embeddings, the authors use WordPiece tokenizer [34] that uses sub-words to represent infrequent or unknown words by breaking them down to smaller pieces. The authors introduce some special tokens, namely, "[MASK]" for masked tokens, "[PAD]" for padding, "[SEP]" to separate different sentences and "[CLS]" for sequence classification. To train the language model, they follow two strategies: Masked language model and next sentence prediction. In masked language model, a percentage of the tokens in the sample are replaced by "[MASK]" token and the model predicts which word should be there. Unlike recurrent neural networks, the transformer architecture allows the model to pay attention in both directions at the same time. In the second task, the model is given two sentences and predicts whether the second sentence follows the first one. The two sentences are separated by the "[SEP]" token and a "[CLS]" token is added as the first token. The prediction is made based on the "[CLS]" token's representation. After pretraining BERT on BooksCorpus [35] and English Wikipedia these two tasks for language modeling, the authors finetune it on different downstream tasks. The authors report superior performance in all tasks.

Hugging Face's Transformers library is a Python library that provides APIs to train and use transformer-based languge models [36]. The library implements various types of transformer-based language models. The library also implements tokenizers used by the models. The users can train their own transformers, share them on the community

hub or download any model from the hub. The library supports GPU accelaration using TensorFlow and PyTorch.

BERTurk is a set of Turkish transformer-based language models trained and shared by Munich Digitization Center at the Bavarian State Library [37]. The model used as the pretrained BERT model in this thesis was the one called "bert-base-turkish-128k-cased" which was trained on the Turkish part of the OSCAR corpus [38], OPUS corpora [39] and another special Turkish corpus, with a vocabulary size of 128000. The total size of the training corpora is around 35GB.

2. DATASETS

One of the main contributions of this thesis is introducing a set of datasets for Turkish for multiple NLP tasks. The source of data is the Turkish Trade Registry Gazette (TRG) published by TOBB. To collect the data, a list of companies were gathered manually, corresponding pages of the gazette were downloaded, image processing was applied and the text was extracted using OCR.

After the text was gathered, the dataset was labeled by 9 annotators. The labels include announcement boundaries, event arguments and event roles.

A parser was written to parse the raw output of the labeling tool to convert to designed dataset formats.

2.1 About the Turkish Trade Registry Gazette

Turkish Trade Registry Gazette is a newspaper published by The Union of Chambers and Commodity Exchanges of Turkey (TOBB) since 1957. Any related announcement companies or courts may have are submitted to TOBB's registry offices. The collected announcements are then published every weekday, usually two days after submission.

While TOBB offers different tiers of subscription plans, anyone can access specific announcements they want using a free account. In the free version, the users can download the page on which the announcement they searched for in image PDF format.

Turkish Trade Registry Gazette is an important resource in many sectors, such as banking, insurance or telecommunication. For example, an non-utilized increase in capital or consecutive closing of branch events may be a negative signal for a bank while they are trying to estimate whether they will be able to collect the credits they gave to a company. On the other hand, opening of branch events around a certain area may present an opportunity for a telecommunication company to get new

subscriptions. Having an up-to-date list of a company's representatives with details may come in handy in situations where money laundering is suspected.

Unfortunately, this valuable information is not kept in a structured format. Even though TOBB has recently started offering searchable PDF format through a premium subscription, not all parts of pages are searchable. In any case, not only is the text unavailable without an extra OCR effort, but also important details can easily be overlooked while an employee is skimming pages of unrelated events or announcements.

Although the Turkish Trade Registry Gazette (TRG) has a unique newspaper format that is only found in Turkey, the information it contains can be found in other sources in many countries. University of Michigan has a list of resources[1] in which one can found corporate filings. Among these, some are not accessible due to paywalls and some already keep the information we are after in a structured format. Among the freely accessible ones, most notably, US Security and Exchange Commission's (SEC) Form 8-K documents resemble TSG announcements closely. Form 8-K are long documents that announce material events such as bankruptcy, disposition of assets or shareholder nominations, which are covered in TRG. Unlike TRG, Form 8-K documents can be found on SEC's Electronic Data Gathering, Analysis, and Retrieval system, more commonly known by its abbreviation EDGAR, [2] platform in digital format and there exists tools to download them easily. Form 8-K documents are loosely structured when compared to TSG, since different event categories have dedicated headers, but the events are described in a free-text format. In Switzerland, there is Swiss Official Gazette of Commerce[3] (SOGC) that publishes similar events every weekday. Issues of SOGC can be downloaded without an account or payment in PDF format and specific announcements can be searched and downloaded in PDF or XML format. The announcements are not as free-formatted as TSG or Form 8-K and are much shorter. However, different from them, SOGC announcements can contain multiple languages. Stock exchange news usually contain similar announcements, examples of which can be seen in London Stock Exchange, Shanghai Stock Exchange and Hong Kong Stock

[1] https://kresgeguides.bus.umich.edu/c.php?g=199820&p=1314096
[2] https://www.sec.gov/edgar/searchedgar/companysearch.html
[3] https://www.sogc.ch/

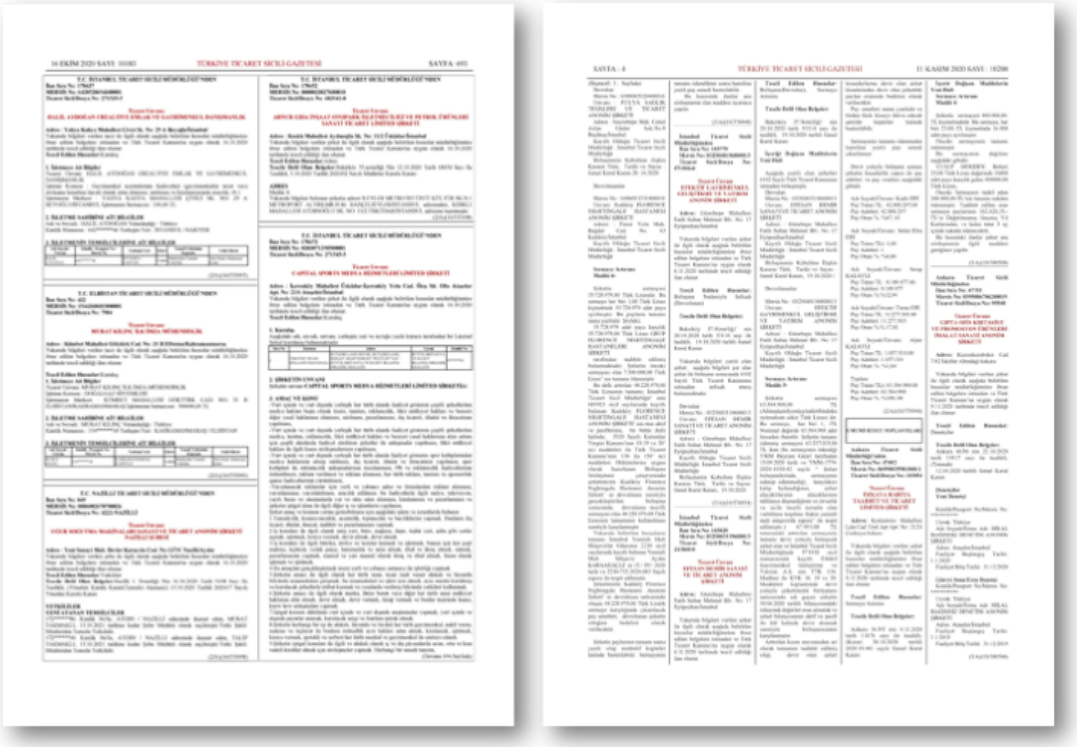

Figure 2.1 : Samples for 2-column and 5-column formats.

Exhange. These can be in PDF format, in which the format of the document also carries information. To the best of our knowledge, no similar resource that requires extensive OCR exists.

Unfortunately, at its current state, information extraction from TRG is a manually-taxing process. Since it has been around for over sixty years, the gazette has went through many layouts. Although the most commonly found layout has a familiar 5-column format found in most newspapers, different 2-column, 3-column and boxed formats also appear. Even within the same issue, many layouts can exist. After August 2020, they discontinued the 5-column format in favor of the 2-column format as seen in Figure 2.1. Although this enabled the TRG to share some information in a table format that is easy for people to read, it also made OCR process harder. The type of information shared in tables is not also standardized. The same information, for example information regarding management, can be shared in a table in one announcement and it can be expressed in natural language in another, even when they are on the same page.

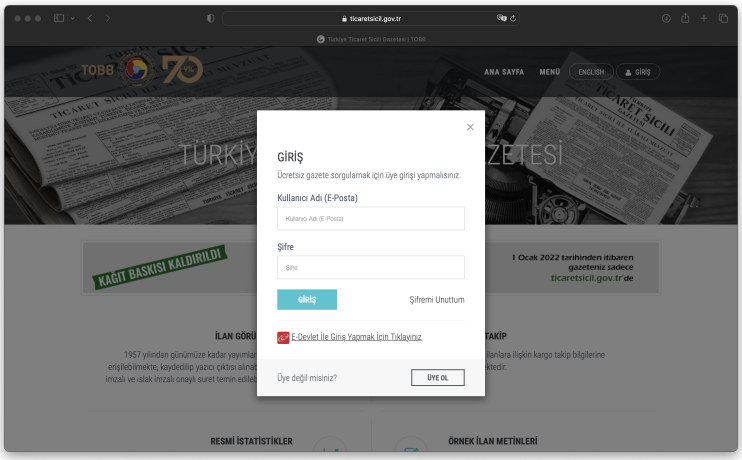

Figure 2.2 : Login page of TRG portal.

Issues of TRG are freely available for anyone with an account via a portal as seen in Figure 2.2. It can be accessed at `https://www.ticaretsicil.gov.tr`.

After entering the portal, the user can search for a specific company's announcements. A whole issue of the gazette is not offered. On the search page, the user is expected to fill some information, as seen in Figure 2.3. The user must select a registry office from the dropdown list, marked by 1, and enter either the registry number, marked by 2 or at least the first five letters of the company's name, marked by 3 in order to search a company. Optionally, the user can apply filters regarding publishing date or court filings, marked by 4. The search results are shown in a table, marked by 5. The columns of the table show registry office information, registry number, company name, publishing date, issue number, page number, announcement type and buttons to purchased signed version of the corresponding pages and to give feedback, in this order. When searching by company name, partial company names can return all companies that begin with the given string, as seen in the second and third rows of the example in Figure 2.3. By clicking on PDF icons, the pages the corresponding announcement is on can be downloaded as a PDF document. It must be highlighted that other announcements in the same page are also present in these pages, as seen in Figure 2.1. TOBB does not offer a service where only the requested announcement can be obtained.

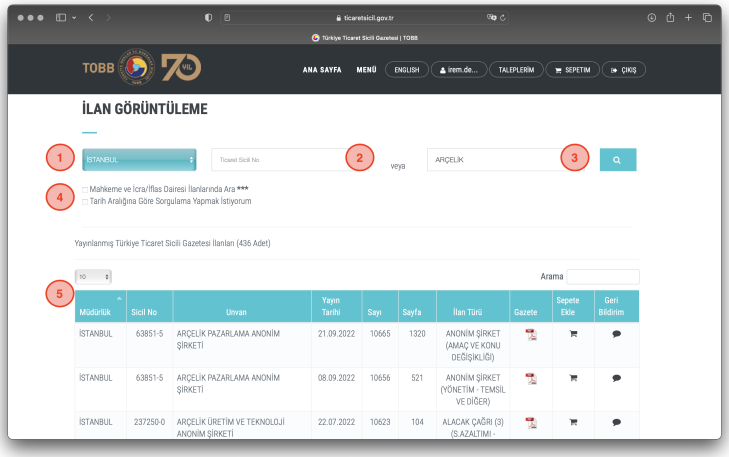

Figure 2.3 : Search page of TRG.

2.2 Data Collection

To understand the announcements better, initially, random announcements of companies from a wide period were analyzed. Since earlier issues of the gazette were published in press, their pages were scanned, as seen in Figure 2.4 and Figure 2.5. Even after digital versions of pages seen in more recent years improved in quality. As seen in Figure 2.6, earlier digitally constructed issues of the gazette do not suffer from salt-and-pepper artifacts like scanned versions, but the resolution is still lower. Thus, we decided to narrow our scope down to issues after 01.01.2014.

Gazette pages were collected in two batches. In the first batch, a list of 161 companies along with registry number and registry office information were manually gathered. During annotation process, another 99 companies were added.

Using this information, a test automation script was used to fill the information and download PDF files automatically. While downloading the files, the search result tables were also downloaded to keep the metadata about the downloaded pages.

The first batch contains announcements from 01.01.2014 to 23.02.2021 and the second batch contains from 01.01.2014 to 06.08.2021. The total number of pages collected by year are as seen in Table 2.1. In terms of years, the number of samples are quite balanced, with the exception of 2021, due to the cutoff date of data collection. Since

Figure 2.4 : A sample page from an early issue of the gazette.

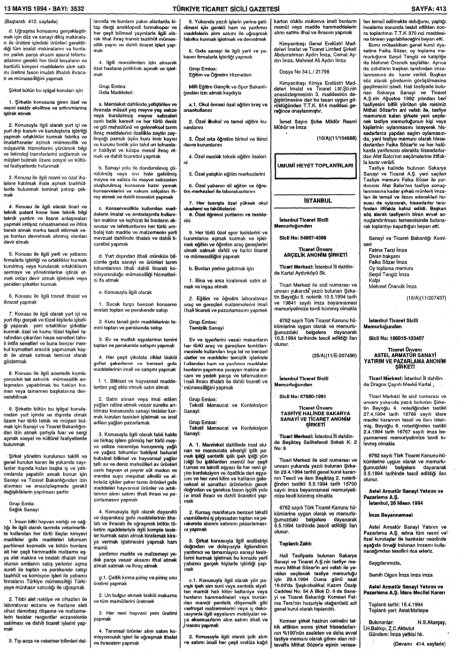

Figure 2.5 : A sample page from a 1994 issue of the gazette.

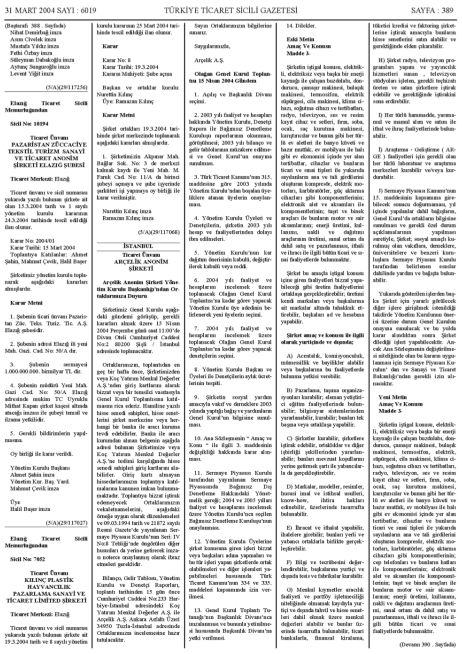

Figure 2.6 : A sample page from a digital issue of the gazette.

the gazette format had recently been changed at the time, 2021 was included to cover the 2-column format as much as possible.

2.3 Text Extraction

After PDF documents were downloaded, a text extraction pipeline was implemented. Since these PDF files are image PDF files, the pipeline contains four steps: extracting images from the PDF pages, image processing, OCR and merging OCR outputs.

Table 2.1 : Number of announcements collected by year.

Year	Number of PDF Documents
2014	401
2015	351
2016	354
2017	380
2018	352
2019	395
2020	410
2021	108

On PDF documents in a given directory, the described pipeline is run. To extract the images, the Python library pdf2image in [40], by Edouard Belval was used. On the extracted images, a series of image processing steps were applied using OpenCV in [41]. For OCR, Tesseract in [29] was used.

Although Tesseract does not have an issue with columned text, it fails when boxes or tables are present. Thus, first boxes in the pages were detected and those larger than some threshold were filtered to detected text boxes and table cells. The remaining boxes were cropped out of the image and the boxes and the remainder of the boxes were OCR'd seperately. Since the coordinates of the cropped parts were known and the XML output of the OCR output also shows the coordinates of the detected text blocks in hierarchy, the outputs could be merged.

The pipeline can be summarized as follows:

- Invert the image.
- Find vertical lines.
- Find horizontal lines.
- Find contours.
- Found bounding boxes.
- Filter bounding boxes by size.
- Mask found table cells.
- OCR the remaining page and get XML output.
- OCR the cropped boxes.
- Sort the box contents according to coordinates of the boxes. Construct rows with the boxes on the same vertical coordinate.
- Merge XML output and boxes:
  - Determine what side of the page the box goes with respect to the middle of the page.

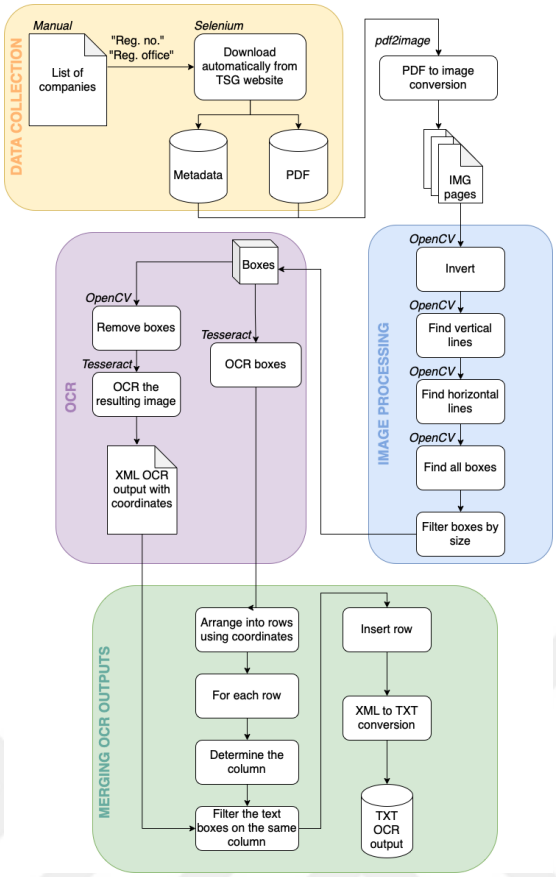

Figure 2.7 : Data collection and text extraction pipeline.

  - Filter ComposedBlock elements in the XML according to the part of the page they are in.
  - Insert the box based on vertical coordinates.

- Convert XML output to text output by traversing the tree and sequentially converting ComposedBlocks to strings.

The flow of the data collection and text extraction pipeline can be seen in Figure 2.7.

Samples for extracted text can be seen in Figure 2.8 and Figure 2.9. As seen in Figure 2.8, the extracted text for a 5-column document contains only minor errors and there is no loss in the data or the order of the text. In Figure 2.9, table information is also extracted while retaining the layout. However, censored ID numbers are hard to extract in both cases. The codes that signify the end of announcement are usually distorted, too.

Figure 2.8 : Sample OCR output for a 5-column page.

Figure 2.9 : Sample OCR output for a 5-column page.

A total of 2973 documents were collected. Based on index information, these documents contain at least 30 different types of announcements. By analyzing how frequently each type occurs and how indicative they are of a companies performance, the following event types were prioritized:

- Composition with creditors
- Notice to creditors
- Change in management
- Capital increase
- Capital decrease

2.4 Data Annotation

Since manual annotation was required, 2973 documents were narrowed down to 489 by whether they were known to contain at least one announcement in one of the prioritized categories.

For data annotation, INCEpTION in [42] was used. INCEpTION is a Java-based web application for data annotation, mainly for NLP tasks. As described in [42], it offers many components for annotation, of which layers and tagsets apply in the case of TRG annotation. The goal of annotation is annotating announcement boundaries and events in announcements. Thus, layers were used to represent events and event boundaries. Features in layers referred to roles in events. For announcement boundaries, there was only a single feature to label the boundary. Tagsets were used to categorize categorical features in labels. For example an entity boundary can either be the start or the end, and after the entity span indicating the boundary was labeled, the corresponding category was selected by the annotators.

Nine annotators annotated the dataset for announcement boundaries and events. Since annotators are expected to annotate data for multiple tasks, the process can be quite challenging. The language used in the TRG is quite formal and hard to understand due to specific terminology. Boundaries of event triggers and arguments can be vague and

how arguments are connected in events can be hard to decode in different cases. To aid annotators during the process, annotation guidelines were provided. Documents were annotated in the order of complexity of the main event contained. First, CC and NTC events were annotated, followed by CIM events, and finally CWC events were annotated. CWC events were considered the hardest, due to the event structure, language used in documents and document lengths.

After one round of annotations, discrepancies in annotations were analyzed and annotation guidelines were updated with corrections. Another round of corrections were applied.

To calculate interannotator agreement, annotators annotated a 382 hand-picked documents that were found to be particularly hard. Cohen's kappa score for interannotator agreement was calculated as 0.91.

2.5 Dataset Creation

As described in INCEpTION documentation in [43], it has a tab-separated output format, named WebAnnoTSV, that provides label definitions and label information of each line as a matrix. At least two columns are reserved for each feature in labels, the columns contain span identifiers and coordinates for each span and the assigned coordinate and coordinate information of connected entities in cases where these apply. Thus, this raw format must be converted to a standardized format.

2.6 Announcement Splitting Dataset

To construct the announcement splitting dataset, each line of the document is extracted from the WebAnnoTSV file with their corresponding feature matrices. Lines with non-empty values in announcement boundary columns are assigned a corresponding label. This way, each line is assigned one of the three categories: start, end, in-between. Training and test datasets were constructed by splitting lines with 4:1 ratio, number of lines in each split and the distribution of classes can be seen in Table 2.2.

Table 2.2 : Announcement splitting dataset statistics.

Category	#Lines - Train	#Lines - Test
Start	1696	424
End	1552	388
In-between	115396	354
Total	118644	29661

2.7 Announcement Classification Dataset

After an announcement splitter model was trained, all announcements in the documents were split. Since the metadata collected in the data collection step refers to a single announcement in the document, it had to be located in order to be used for document classification.

By searching the registry number in the metadata in the split announcements, the metadata was matched with the announcement text. Since the metadata contains announcement types, an announcement type was also assigned to matched announcements. Since the same announcement type in the TRG can appear in multiple forms depending on the type of the company, the announcement types that refer to the same event where unified. After this operation, 15 most common announcement types and one that refers to all others were considered in the label set. The number of announcements in the dataset for each announcement type can be seen in Table2.3.

Table 2.3 : Announcement types in announcement categorization dataset.

Original Categories	Merged Categories	Counts
Şube (Yönetim - Temsil), Anonim Şirket (Yönetim - Temsil ve Diğer)	Management - Representation (MR)	1032
Genel Kurul Toplantıya Çağrı	Call to Meeting (CtM)	274
Anonim Şirket (Tadil)	Amendment (Tadil) (Amend)	205
Konkordato Alacaklı Top., Konkordato Tasdiki	Composition with Creditors (CC)	202
Alacaklılara Çağrı (3)	Notice to Creditors (NTC)	180
Şube Kapanış	Branch Closing *(BC)*	39
Limited Şirket (Pay Devri)	Share Transfer (ST)	31
Genel Kurul İç Yönergesi (TTK - M.419)	Internal Directive (İç Yönerge) (ID)	41
Limited Şirket (Sermaye Artırımı)	Capital Increase (CI)	18
Limited Şirket (Kuruluş), Anonim Şirket (Kuruluş), Gerçek Kişi Ticari İşletmesi (Kuruluş), İş Ortaklığı İşletmesi (Kuruluş)	Foundation (Foun)	15
Anonim Şiret (Amaç ve Konu Değişikliği)	Change of Subject (CoS)	6
Sermaye Azaltımı	Capital Decrease (CD)	6
Bölünme	Demerger (Demer)	6
Limited Şirket (Adres Değişikliği), Anonim Şirket (Adres Değişikliği)	Change of Address (CoA)	35
Birleşme	Merger (Mer)	23
All others	Other	475

Table 2.4 : Description of the roles of Person entity.

Role	Type	Description
Name	String	Name of the person.
ID Number	String	ID number and its type. Different ID numbers are given for Turkish nationals and foreigners.
Nationality	String	Nationality or the country of origin.
Address	String	Address of the person in long or short form.

2.8 Event Extraction Dataset

The event extraction dataset contains labels for event triggers, arguments and roles. Four event categories are defined to cover the most important and frequent events. Composition with Creditors (CC) ("Konkordato" in Turkish) event covers concerning composition, such as the start, extension or termination of a composition term. Notice to Creditors (NTC) ("Alacaklılara Çağrı" in Turkish) event covers the cases where the creditors are required to act due to a financial difficulty a company faces. Lastly, capital increase and capital decrease events are represented by a single event type, Change in Working Capital (CWC).

These events were represented as templates where arguments are connected to a trigger. Some events require complex event arguments that have arguments of their own, like persons or amounts, referred to as auxiliary entities. For auxiliary entity templates, the main argument is the trigger. This structure makes the TRG event extraction dataset one that contains n-ary and nested structures. For the remainder of this chapter, in auxiliary entity and event templates, triggers are shown in bold and required arguments are in italics.

CIM events require four auxiliary entities. These auxiliary entities are Person, Money and Authorization Type.

The Person entity represents a person with their personal information if given. The descriptions of the roles of Person entity can be seen in Table 2.4.

The CIM event always specifies the title concerning the change. Thus, the auxiliary entity Title is introduced. Its roles can be seen in Table 2.5.

Table 2.5 : Description of the roles of Title entity.

Role	Type	Description
Title name	String	Name of the title.
Title holder(s)	Person	The holder(s) of the title.
Valid from	String	The date the title is valid from.
Valid until	String	The date the title is valid until.
Duration	String	The date the title is valid for.

Table 2.6 : Description of the roles of Money entity.

Role	Type	Description
Amount	String	The amount along with its currency and written form (when available).
Number of shares	String	The total number of shares the amount corresponds to.
Price per share	String	The price of a single share.
Shares	Amount	Distribution of the amount among shareholders.
Shareholder	Person	If the Money entity is used to describe some share, the shareholder must be specified.

The Money entity is used to represent monetary amounts found in CWC events. In order to cover distribution of monetary amounts, this entity has recursive arguments. The descriptions of the roles of Money entity can be seen in Table 2.6.

The Money entity is a generic entity that can represent a capital amount, as well as its components, like the share price or how the amount shares corresponds to. When it represents a share price, change amount or old capital, all roles except for Amount are left empty. In the case it represents a share, the share price can be present if it appears in the context.

CIM events require the Authorization Type. This entity shows whether the given authority is shared with others, and if it is, with whom. The descriptions of the roles of Authorization Type entity can be seen in Table 2.7.

In Composition with Creditors (CC) event, bankrupt companies share announcements about agreements they made regarding their debts. The announcement states whether a composition term was given, extended or cancelled. Announcements with CC events usually cover a single event and have shorter text. The descriptions of the roles of CC event can be seen in Table 2.8.

Table 2.7 : Description of the roles of Authorization Type entity.

Role	Type	Description
Type	Category	The type of authority given, namely, "severally", "jointly" or "limited".
Shared with	Person	The people with whom the authority is shared in joint cases.

Table 2.8 : Description of the roles of Composition with Creditors event.

Role	Type	Description
Action	Category	A string represented by one of the three categories, namely, "given", "extended" or "cancelled". Describes the action taken in the event regarding the composition term.
Term	Category	A string represented by one of the three categories that represent the type of term the action was taken for, namely, "temporary term", "final term", "temporary/final term".
Valid from	String	The date the term is valid from.
Valid until	String	The date the term is valid until.
Duration	String	The duration the term is valid for.

Table 2.9 : Description of the roles of Notice to Creditors event.

Role	Type	Description
Cause	Category	A string that represents the why the notice has been issued, namely, "due to merger", "due to demerger", "due to liquidation".
Announcement Number	String	This type of announcements are published for three times before the creditors can be held responsible for not taking action. The time arguments usually refers to this number.
Valid from	String	The date starting from which the creditors should take action from. It can be an expression that refers to the announcement number and publishing date, registration date of the announcement or a regular date expression like 12.12.2020.
Duration	String	The duration the term is valid for.
Address	String	The address the creditors should apply to.

In the case where the term is cancelled, the date roles are not expected. In other cases, either "valid from" and "duration", "valid until" and "duration", "valid until" or "duration" are expected.

In an Notice to Creditors (NTC) event, the companies issue a 5-column format creditors to collect their credits due to a past merger, division or liquidation event. They specify the dates between which the creditors should take action. Like CC events, most announcements that contains this event type are shorter and contain only a single event. The roles found in this event and their descriptions can be seen in Table 2.9.

An important characteristic of NTC events is the complex time expressions they contain. An NTC event should be published three times before it is put into effect. Since an NTC event happens because of a merger, demerger or liquidation, the registration date of these events may be given explicitly or they may be referred implicitly. In both of these cases, "valid from" refers to the registration date. If the registration date of the cause is implicitly given, it should be the "valid from" argument, otherwise the expression that refers to the cause should be marked as such.

Change in Working Capital (CWC) events announce an increase or decrease in the working capital. These are usually found in long announcements that contain multiple events. Additionally, the announcements may contain other events with similar roles,

Table 2.10 : Description of the roles of Change in Working Capital event.

Role	Type	Description
Change	Category	The type of change, namely, "increase" or "decrease".
New Capital	Money	The amount of capital after the change.
Old Capital	Money	The amount of capital before the change.
Change Amount	Money	The amount of which the change was made.

Table 2.11 : Description of the roles of Change in Management event.

Role	Type	Description
Action	Category	The action taken on the title, namely, "appointment", "removal" or "extension".
Title	Title	The title the action was taken on.
Auth. Type	Auth. Type	The authorization type given with the title.

like share distribution, that are not covered in this dataset. The context the event is in may contain irrelevant money amounts and persons. The roles of the CWC event can be seen in Table 2.10.

The same CWC event can be repeated in the announcement as a summary or in full. Since companies may announce multiple events at once, conflicting events may be present in the same announcement. A long list of shareholders may be listed. In older documents, amounts may have OCR errors in them.

Change in Management (CIM) events show when a person is assigned to or removed from a managerial position or when their service duration is extended. CIM requires Title and Person entities and may also include Authorization type. This dependency makes CIM a nested event. Description of the roles in this event type can be seen in Table 2.11.

In announcements that contain CIM events, there may be multiple events with different actions. Within the same announcement, the same person can have a multiple titles on which changes are made. Mostly in recent announcements, due to personal privacy laws, ID numbers are fully or partially masked, which introduces OCR errors.

Table 2.12 : Number of sentences by types of events contained in the documents.

Event Types Present	#Sentences	#Documents	Avg. #Sentences
CIM	2854	297	9.61
CIM, CWC	1028	21	48.95
CWC	2676	167	16.02
CC	1268	305	4.16
NTC	518	489	1.06
NTC, CWC	5	5	1.00

Sometimes, authorization type can be stated after a long explanation of its extent, which moves it far from the trigger.

Example event annotations are illustrated in Figure 2.10. As seen, depending on the type of events contained in the announcement, event arguments can spread over sentences, multiple events occur in each others vicinity and complex relations can be present. These make data annotation phase quite challenging.

2.8.1 Dataset statistics

The event extraction dataset consists of 1284 documents. The dataset has a total of 469264 tokens and 8349 sentences, however these are not distributed evently. Some documents also contain events of multiple types. Statistics for number of events contained in documents and the number of sentences in each type can be seen in Table 2.12. As seen, documents that contain simpler events like NTC tend to be much shorter, but the documents can be much longer.

Figure 2.11 shows how individual documents differ in terms of sentence and argument count by event types. Documents that contain CIM and CWC events are diverse in terms of both sentence count and argument counts as they can contain multiple events. Documents that contain NTC and CC events are much simpler by both metrics.

Table 2.13 shows the number of triggers and arguments and Table 2.14 shows number of relations between triggers and arguments and their types. As seen in Table 2.14, roles do not appear equally even for the same trigger type, since some arguments can be omitted in the events. In some cases, the same trigger can be in different roles with the same type of argument, like in the case of Title “Name”, “Valid until” and “Valid from”.

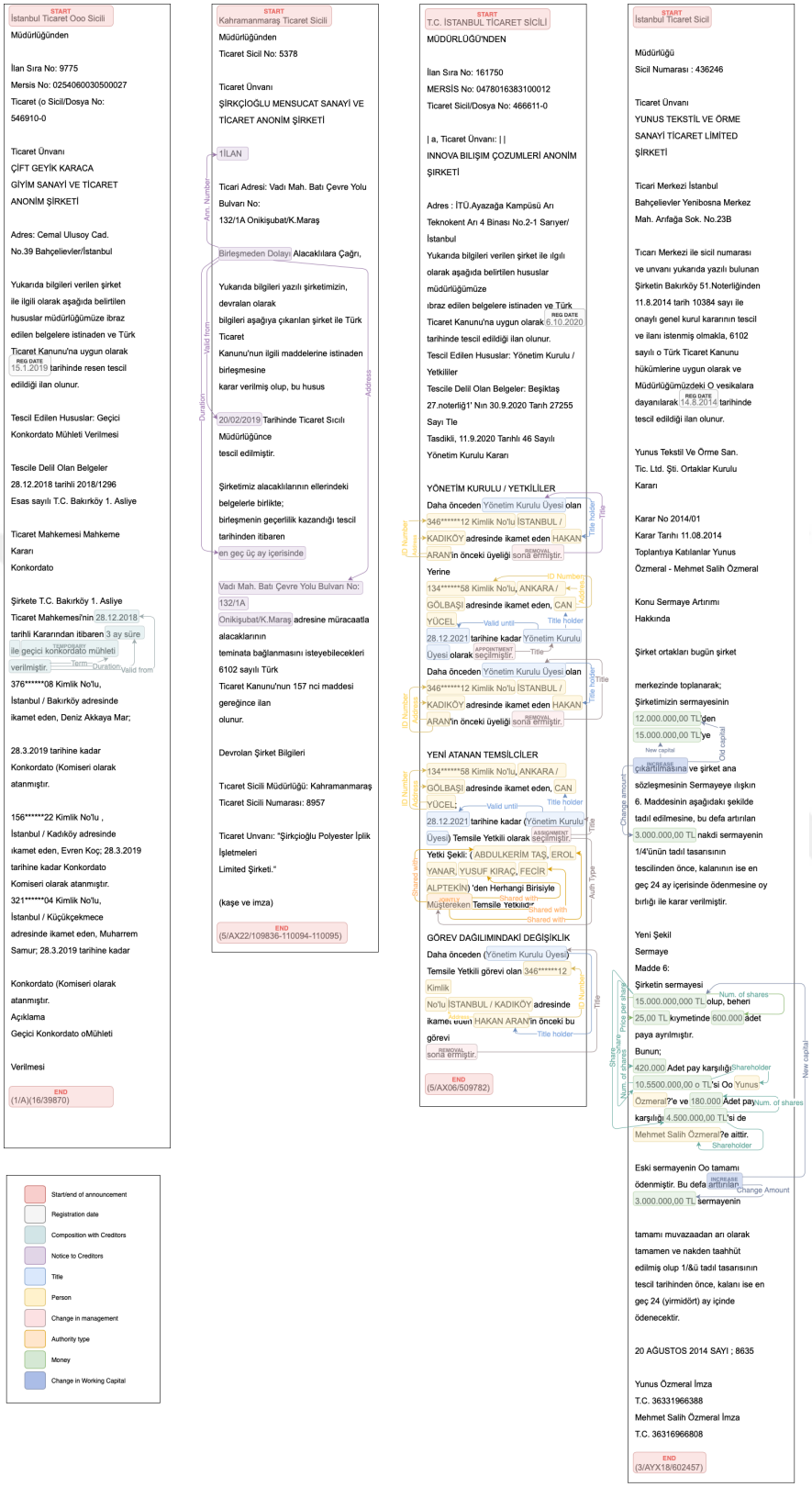

Figure 2.10 : (Left to right) Announcement samples for CC, NTC, CIM and CWC events.

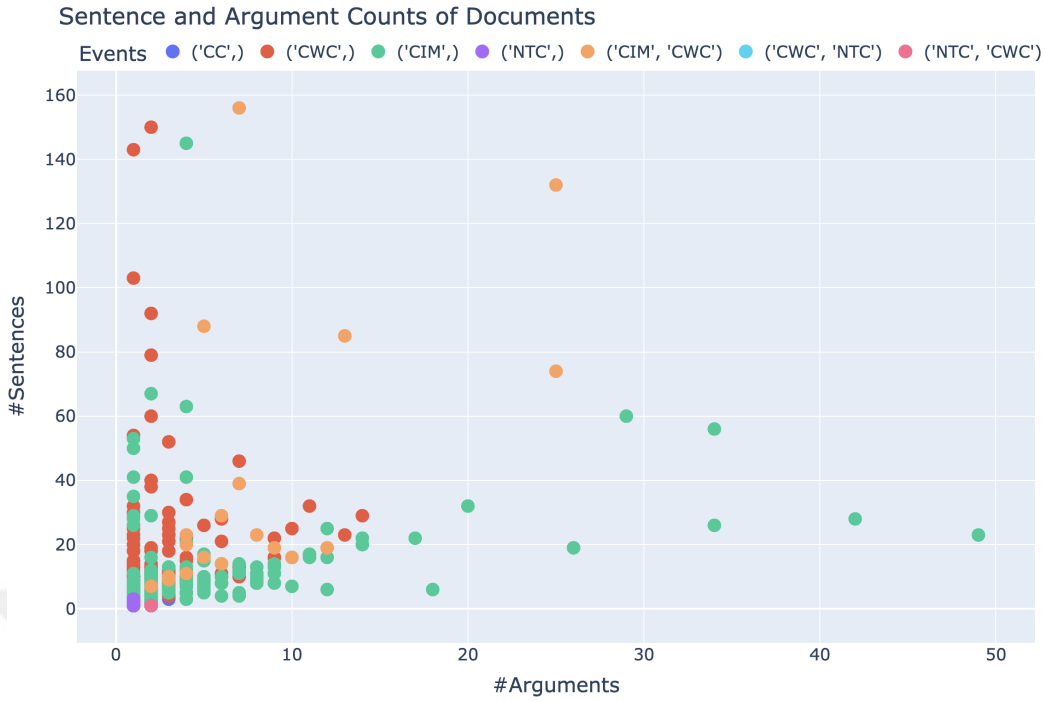

Figure 2.11 : Scatter plot of sentence and argument counts for individual documents. One document with more than 200 sentences and one document with 70 arguments are excluded.

Table 2.13 : Event argument and trigger counts by type

		Count
Triggers		
CIM	action	976
CWC	change	307
CC	action	417
NTC	cause	494
Arguments		
AuthType	type	396
CC	term	415
General	date	1979
General	address	1344
General	duration	1192
Money	amount	1260
Money	num. shares	199
NTC	ann. number	494
Person	ID. number	840
Person	name	1331
Person	nationality	174
Title	name	1043

Table 2.14 : Role counts between triggers and arguments by event type.

Event	Role	Count
CIM	action	1258
CIM	auth. type-shared with-name	240
CIM	auth. type-type	561
CIM	title-duration	455
CIM	title-name	1258
CIM	title-title holders-address	999
CIM	title-title holders-ID number	972
CIM	title-title holders-name	1258
CIM	title-title holders-nationality	220
CIM	title-valid from	39
CIM	title-valid until	410
CWC	change	489
CWC	change amount	389
CWC	new capital-amount	481
CWC	new capital-num. shares	361
CWC	new capital-share price	354
CWC	new capital-shares-amount	317
CWC	old capital-amount	310
CC	action	417
CC	duration	352
CC	term	415
CC	valid from	302
NTC	address	493
NTC	ann. number	494
NTC	cause	494
NTC	duration	492
NTC	valid from	481

2.8.1.1 Comparison with other datasets

In terms of its structure and content, ChFinAnn is the most similar dataset to the TRG event extraction dataset. ChFinAnn is a document-level financial event extraction dataset introduced by [1]. The authors use distant supervision to annotate six events containing six to nine arguments. Since the authors model the problem of entity extraction as table filling, the events do not have triggers.

ChFinAnn is a larger dataset with 40146 documents in total thanks to the automatic labeling approach, as opposed to TRG-EE's 1284 manually labeled documents. In ChFinAnn, events can most frequently span 5 sentences or 7.49 sentences on average and 62 sentences at maximum. While ChFinAnn's events span a larger distance within the document, over 53% of events span more than one sentence in TRG. Of the four event types considered in TRG, CC and NTC are known to be easier and tend to occur at the sentence-level.

$$\text{\#arguments scattering} = \frac{\text{\#event mentions}}{\text{\#arguments}} \tag{2.1}$$

where number of event mentions refer to the number of sentences with event occurrences in the document.

[19] defines argument scattering as seen in Equation 2.1. TRG's arguments scattering measure ranges from 0.002 to 0.5. This value is between 0.05 and 1.33 for ChFinAnn. In addition, time expressions in ChFinAnn are very standard, in a formal year-month-day form. However, they can get very complex in TRG. It should also be noted that the arguments in TRG can get even more complex due to OCR errors and natural language expressions such as durations like "until the third announcement".

Comparison of the TRG dataset with other datasets mentioned in 1.1 can be seen in Table 2.15 and Table 2.16. In both tables, TRG-EE refers to the TRG event extraction dataset and TRG-AC refers to the announcement classification dataset. For convenience, the task of classifying a multi-sentence document is referred to as document classification, abbreviated with DC. Kleister-NDA and Kleister-Charity

datasets [5] are similar to the TRG dataset since they also work with OCR'd PDF documents. DCFEE [6], DuEE-fin [23] and Commodity News [8] datasets are also similar since they are financial or commodity event extraction datasets. Trade the Event [6] dataset is a financial event detection dataset that is similar to the TRG argument classification dataset. Despite Protest-Event dataset [27] being a protest-news dataset, it also has a text classification subset and an event extraction subset. Finally, the Transaction dataset is an information extraction dataset that resembles an event extraction dataset with a single event, where the event is a money transfer. Overall, the TRG dataset differs in that text is extracted from a gazette using OCR and events have complex structures that include auxiliary events, unlike any other event extraction dataset. The TRG event extraction dataset is also quite rich in terms of the number of arguments it contains.

Table 2.15 : Comparison of various datasets based on their content and availability.

Name	Language	Domain	Source	Public	OCR?	Task
Kleister-NDA [5]	English	NDA information	EDGAR	yes	yes	KIE
Kleister-Charity [5]	English	Charity information	UK Charity Commission	yes	yes	KIE
Transaction [28]	Turkish	Transaction information	Banking documents	no	yes	IE
Trade the Event [6]	English	Organizational events + Financial events	Online news websites	yes	no	ED
ChiFinAnn [1]	Chinese	Financial events	Online news websites	yes	no	EE
DCFEE [14]	Chinese	Financial events	Stock exchange news	yes	no	EE
Commodity News [8]	English	Financial events	Online news websites	yes	no	EE
Protest-Event Dataset [27]	English	Protest event detection	News articles	yes	no	DC
Protest-Event Dataset [27]	English	Protest event detection	News articles	yes	no	ED
Protest-Event Dataset [27]	English	Protest event detection	News articles	yes	no	EE
DuEE [24]	Chinese	Real-life news	Baijiahao news	yes	no	EE
DuEE-fin [23]	Chinese	Financial news	Financial announcements, news and judgements	yes	no	EE
TFEEC [26]	Turkish	Stock exchange news	Online finance news articles	yes	no	EE
TRG-EE	Turkish	Organizational events + Financial events	Turkish Trade Registry Gazette	yes	yes	EE
TRG-AC	Turkish	Organizational events + Financial events	Turkish Trade Registry Gazette	yes	yes	DC

Table 2.16 : Comparison of various datasets based on structure.

Name	Argument Types	Event types	Document level?	Annotation Method	#Documents	#Sentences	#Entities
Kleister-NDA [5]	4	N/A	yes	manual	540	N/A	2160
Kleister-Charity [5]	8	N/A	yes	semi-automatic	2778	N/A	21612
Transaction [28]	14	N/A	yes	manual	3500	N/A	47846
Trade the Event [6]	N/A	11	yes	manual	9721	N/A	N/A
ChiFinAnn [1]	23	5	yes	distant supervision	32040	629338	341663
DCFEE [14]	5	4	yes	distant supervision	2976	35980	38477
Commodity News [8]	21	18	no	manual	150	N/A	12800
TFEEC [26]	15	25	no	manual & automatic	34746	323945	697463
Protest-Event Dataset [27]	N/A	N/a	yes	manual	9548	N/A	N/A
Protest-Event Dataset [27]	N/A	N/A	no	manual	2432	22443	N/A
Protest-Event Dataset [27]	6	6	no	manual	896	3468	9122
DuEE [24]	121	65	no	manual	11224	16956	61160
DuEE-fin [23]	92	13	yes	manual	11645	N/A	97481
TRG-EE	29	5	yes	manual	1284	8349	11818
TRG-AC	N/A	16	yes	manual	2588	27804	N/A

3. ANNOUNCEMENT CLASSIFICATION

As described in Chapter 2, multiple announcements from different companies can be found on the same page of the TRG. While accessing a specific announcement, the whole page is retrieved, thus, the targeted announcement should be extracted from the page. Moreover, although the companies of surrounding announcements are unknown, these announcements may contain valuable information for other tasks, such as event argument extraction and event extraction.

An announcement usually starts with a mention of the registry office they were submitted to or the name of the court that submitted it, however in more rare cases, it can start with other information, like the company's trade registry number. Announcements always end with a alphanumeric code in one or more pairs of parentheses. Although announcement boundary splitting looks like a simple problem that can be handled with handcrafted rules and regular expressions, it requires a learning-based approach. The types of information mentioned for the start of an announcement can be also appear in other parts of the announcement. For example, the announcement may be about the company's trade registry change and the company's new registry office can be given in a separate line. As for the code at the end, it is the part of the announcement OCR errors are introduced most frequently. For example, the OCR tool can mistake two parenteses that are next to each other (")(") for the letter X or interpret numbers as words ("ASLD', "451)"). Since announcement boundaries were annotated, a learning-based approach was found faster and more effective.

Once announcement boundaries are detected and announcements on the page are split, they can be classified. In the TRG, announcements are not usually strictly grouped by registry offices or announcement types. Since the metadata collected from the TRG website contains information about a certain announcement on a page, a document classifier can be trained after matching the information with the isolated

announcement. As described in Chapter 2, an announcement classification dataset was constructed based on this method and announcement types were unified.

Announcement classification comes with a set of challenges. Announcements are usually long documents, which are known to be hard to handle especially for transformer models. Moreover, announcements of different types may contain similar language. These sentences can be also be quite long. Longer contexts can be helpful for classification performance, but the length of the document also determines the processing time. Thus, finding a good spot in the processing time and performance trade off is also an important aspect of the problem to consider.

The work presented in this chapter can be found in [44].

3.1 Announcement Splitting

The OCR output of the TRG is line-based to reflect the structure of pages better. As mentioned in Chapter 2, during the annotation phase, this layout was preserved to help annotators when they needed to switch between the text and PDF versions. Thus, announcement boundaries were defined as the first or the last line of the announcement. The goal of the announcement splitter is to assign each line one of the following three categories: Start, end, in-between.

In line with this goal, a transformer-based model was used. A classification token, "[CLS]" was added to the beginning of lines. Then these were fed into the BERTurk model introduced in [37] and the representation for the "[CLS]" token was forwarded to a linear layer for classification.

Announcement splitting problem is an imbalanced classification problem since most lines are not boundary lines. Thus, it is important to select an appropriate performance metric. In such cases, accuracy paints an overly optimistic but false picture, since assigning the majority class to all samples would give an accuracy of 0.97 in the case of this problem. Thus, precision, recall and F1 scores were calculated. Shortly, for a given class, precision is the percentage of samples correctly predicted by the model, recall is the percentage of positive samples found by the model and F1 is their harmonic mean. Performance of the announcement splitting model can be seen in Table 3.1. The

Table 3.1 : Performance of the announcement splitting model on the test set.

Class	Precision	Recall	F1	Support
Start	0.95	0.92	0.94	424
End	0.98	1.00	0.99	388
In-between	1.00	1.00	1.00	28849

model is able to distinguish in-between lines from boundaries and end lines. Although start lines were more challenging for the model, it was able to classify them quite well, as reflected by the F1 score of 0.94.

This model was used to create the announcement classification dataset as mentioned in Chapter 2. Announcements were split based on end lines and a line of announcement information was matched with split announcements by searching the registry number. Registry number information was observed to be preserved and the operation was carried out with an accuracy of 0.97.

3.2 Announcement Classification

Announcements are longer pieces of text consisting of sentences, thus, this problem can be considered a document classification problem. Although transformer-based language models are very popular in NLP, most models can only handle a set number of tokens and the performance worsens as the context gets longer. For example, the transformer-based language model used in all models in this thesis, BERTurk in [37], can only handle a maximum of 512 tokens. Since announcements are similar in terms of wording, document-level context should be incorporated into models.

As the context gets longer in terms of both the number of tokens and the number of sentences, the model usually has to perform more operations, leading to increased training and prediction times. Thus, for this problem, we consider this aspect of the problem by the same model with different context sizes.

Figure 3.1 shows number of sentences in the announcement classification dataset. As seen, most documents contain fewer than 20 sentences, but the number of sentences can be much higher.

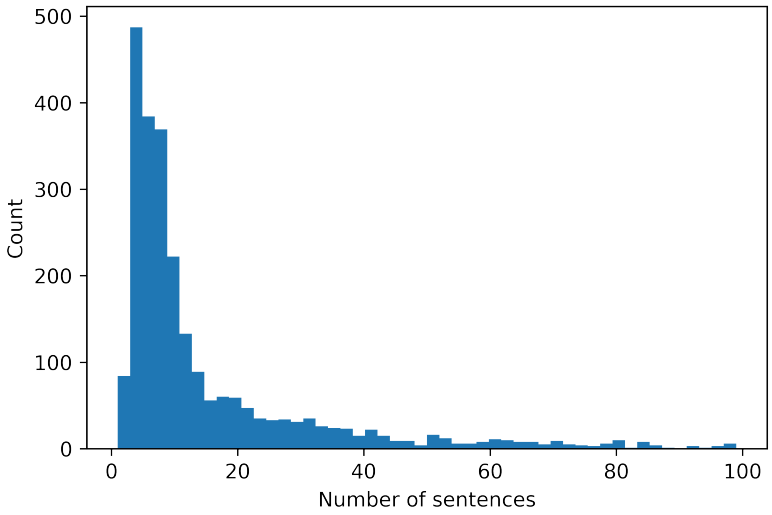

Figure 3.1 : Number of sentences in the announcement classification dataset. 42 announcements with more than 200 sentences are excluded.

For this problem, sentence length was limited to 200 and longer sentences were broken. As seen in the histogram of token counts for sentence in Figure 3.2, sentences can be quite long. Considering these sentences would be tokenized by the transformer model's tokenizer, these numbers would increase drastically.

For this reason, we train models on 5-sentence, 10-sentence and 25-sentence contexts. For the first two, sentence are limited to 200 tokens and for the last one, sentences are limited to 100 tokens.

To produce context-aware embeddings for tokens, BERTurk model in [37] was used. The model was loaded using Hugging Face's transformers library in [36]. The model architecture can be seen in Figure 3.3. All sentences in the document are fed to BERT to produce word representations. Max-pooling operation is applied over the token representations to produce sentence representations. The sentence representations are then stacked and fed to a linear layer and max pooling is applied over the sentence representations to produce the document representation vector. The vector is passed to a final linear layer for classification.

Since this is a classification problem with 16 classes, cross entropy loss is used. Adam optimizer with a learning rate of 2×10^{-5}, β_1 of 0.9 and β_2 of 0.99 is used. The training is carried out in 5 folds.

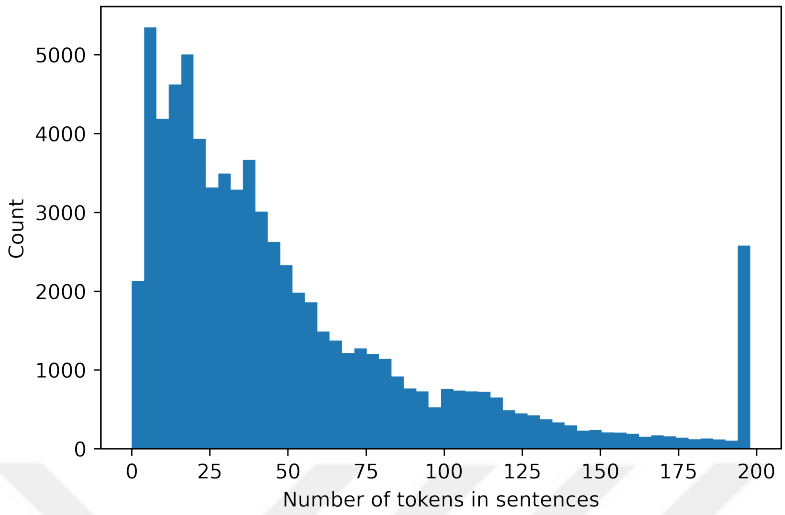

Figure 3.2 : Number of tokens in sentences in the announcement classification dataset.

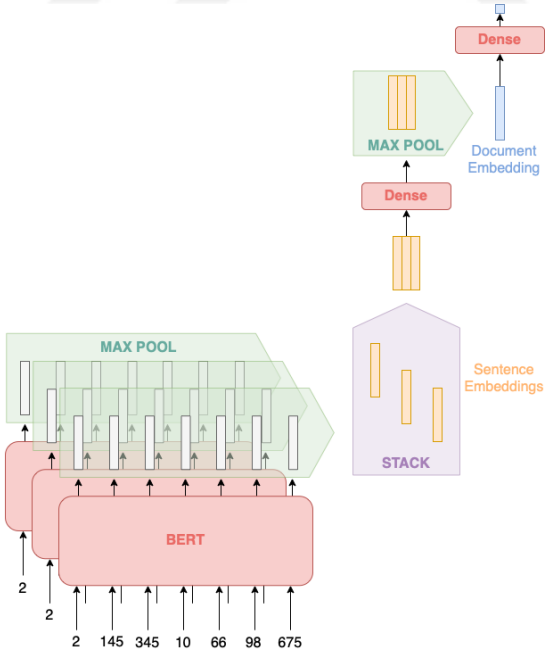

Figure 3.3 : Network architecture for announcement classification model.

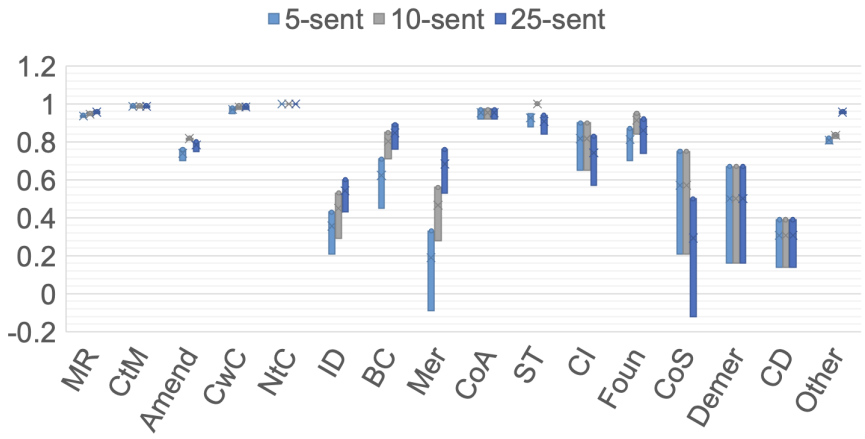

Figure 3.4 : Performance of the announcement classification network in terms of F1 score over 5 folds.

To evaluate the performance of the model, precision, recall and F1 are calculated. A bar chart depicting F1 scores can be seen in Figure 3.4. The classifier has a macro F1 score of 0.83. The model is able to classify more frequent announcements correctly and reliably in all cases. In the cases where there are sufficient number of samples, the performance tends to get better as the context gets longer. However, the model cannot generalize well for infrequent categories, indicating a need for improvements on the dataset side for the future.

The inference time increases linearly with context size. 5-sentence, 10-sentence and 25-sentence models take 5 minutes 22 seconds, 9 minutes 1 second and 25 minutes 44 seconds over the whole dataset for prediction.

4. EVENT EXTRACTION

4.1 Problem Definition

As defined in the ACE 2005 annotation guideline, an event is described as a change of state involving various parameters at a specific time and place [45]. It can be represented as a schema that is composed of event triggers and arguments where the trigger indicate a change of state and arguments define the parameters of the change. The problem of event extraction focuses on transforming this unstructured information found in text into a predefined, structured format.

As mentinoned in [46], event extraction terminology defines the following:

- **Event trigger:** The parameter of the event that indicates the event, usually expressed by a verb.
- **Event argument:** General or event-specific entities that define the parameters of the state change.
- **Argument role:** The relation between the trigger and argument that defines how the argument takes part in the event.
- **Event mention:** The part of text that contains an event.
- **Event record**: The instance of an event as represented in the predefined structured format. An event mention or trigger can produce multiple event records depending on design.

Three main approaches are commonly found in the literature in the context of event extraction. In the first one, the problem is handled in multiple steps. These steps can include event detection, where the event mentions in the text are detected. Event trigger extraction, where the trigger expression in the text is extracted. Event argument extraction, where the arguments are extracted. Event role extraction, where event the

relations between the triggers and arguments are predicted. Based on design, some steps can be eliminated or merged. Event argument extraction and trigger extraction are similar to NER and can be handled in a single step.

In the second approach, the problem can be converted to a table-filling problem. As preferred by [1], with this approach, triggers can be eliminated and roles can be assigned to arguments directly.

In the third approach, the problem is modeled as a question-answering problem. As an example approach, in [12], the authors use two questions, one asking what the role is and another asking what plays the role. These questions and the context are encoded using two separate encoders and the arguments and their corresponding roles are extracted jointly using argument and role classifiers. In [47], the authors use a BERT-QA model and optimize questions regarding arguments and triggers by designing hand-crafted question templates.

In this thesis, the problem is handled as a table filling problem with two steps. In the first step, triggers and arguments are extracted following an approach similar to NER. In the second step, event roles are extracted to produce event records.

4.2 Event Trigger and Argument Extraction

Event trigger and argument extraction refer to the extraction of token spans that correspond to triggers and arguments respectively. This problem is a token classification problem similar to NER, but the goal of extracting only the entities that are involved in an event distinguishes this problem. In this work, event trigger and argument extraction were modeled as token classification problems and BERT-based models were used to obtain token representations. Experiments were carried out to observe the effect of IOB tags, an additional CRF layer on top of BERT and separate extraction of triggers and arguments. After the experiments, the approach that yields the best results were chosen before proceeding to event role extraction.

The language model used to obtain token representations is a BERTurk model introduced in [37], finetuned on TRG-EE data. In preprocessing, OCR errors were corrected by mapping invalid tokens to their valid counterparts in the model's

Table 4.1 : Architecture of the base token classification model.

Layer
Linear(in_features=256, out_features=num_labels)
Dropout(0.1)
Linear(in_features=768, out_features=256)
Dropout(0.1)
BERT

vocabulary using a dictionary and numeric strings where replaced by a mask that shows the number of digits they contain (ex. “34” to “2d”, “2345” to “4d”). Since the OCR input does not preserve sentence boundaries, sentence tokenization was applied as well.

For all experiments, a base token classification model was shared. The architecture of this base model can be seen in Table 4.1.

Initially, the base model was trained using IOB labels. Since performance of the second stage of event extraction depends heavily on the successful extraction of triggers especially, experiments were carried on to observe how both trigger and argument extraction performance could be improved.

The experiment settings can be defined as:

- **IOB/no-IOB** Whether IOB tags were used while predicting trigger and argument expressions.
- **CRF/no-CRF** Whether an additional CRF layer was added on top of the last linear layer.
- **one-stage/two-stage** In the one-stage setting, a single network is used to predict both trigger and argument expressions, while the two-stage model uses two separate instances of the same model to predict them separately.

The motivation behind observing these variables was seeing how they affect performance in a low-data settings with a high number of labels, since there are 6 trigger types and 15 argument types in TRG-EE. Although using IOB tags to distinguish entity boundaries is preferred in NER, it doubles the number of labels

and increases the model complexity. Since the same type of entities do not usually appear next to each other in TRG, experiments were carried out to see if they could be omitted. The CRF layer is a common choice to correct labels predicted for tokens by using predicted labels for the whole sequence, thus, in these experiments, a CRF layer was added on top of the base token classification model to see whether the predicted entity boundaries and labels assigned by the model (such as invalid B-I transitions) could be improved. Lastly, since the TRG uses triggers to define the event, event extraction can only get better with better trigger extraction. To test whether using two identical models that focus on extracting triggers and arguments separately improves trigger extraction especially, two copies of the same model were trained and their effect on the performance was observed.

First, the effect of IOB tags and the added CRF layer were compared on one-stage models. Based on the experiment results, the selected settings were further applied to a two-stage model to decide the final model design.

4.2.1 Evaluation metrics

Since this subproblem resembles NER closely, CoNLL 2003 NER metrics, defined in [48], were used. [48] uses precision, recall and $F_{\beta=1}$, calculated as shown in Equation 4.1, Equation 4.2 and Equation 4.3 respectively.

$$precision = \frac{\text{\#correctly found entities}}{\text{\#found entities}} \tag{4.1}$$

$$recall = \frac{\text{\#correctly found entities}}{\text{\#entities in corpus}} \tag{4.2}$$

$$F_{\beta} = \frac{(\beta^2 + 1) * precision * recall}{(\beta^2 * precision + recall)} \tag{4.3}$$

Since the dataset is quite small, setting fixes training, validation and test sets could yield different ratios for different entities in these sets. Thus, all models were trained using 5-fold cross-validation and the macro and micro averages for these performance

metrics and the standard errors were calculated. The same approach was applied while reporting the overall performance as well.

4.2.2 Experiment results and discussion

4.2.2.1 The effect of using IOB tags

To observe the effect of using IOB tags, the first two and last two columns in Table 4.2 can be compared within themselves. As seen in the first two columns, at the absence of CRF, IOB tags provide better recall, but worse precision, both in terms of macro and micro scores. In terms of macro and micro F1 scores, IOB labels provide only slightly better performance. In the latter two columns, the effect of using IOB tags is observed. In this setting, in both macro and micro scores, IOB tags yield worse performance in terms of recall, but better performance in terms of precision. This, in return results in a better performance without IOB tags in terms of F1 score. However, it is important to note that the overall performance turns out to be similar in terms of both micro and macro F1 score when IOB labels are used without a CRF layer.

Same comparison can be done on Table 4.3. By comparing the average micro F1 scores of trigger and argument detection, it can be seen that not using IOB tags yield better performance with or without CRF. Moreover, when performance is compared at the trigger- and argument-level, in models without a CRF layer, not using IOB tags yields better scores in three out of four triggers and six out of twelve arguments. In models with a CRF layer, not using IOB tags yields better performance for all triggers and five arguments out of twelve.

Since not using IOB tags yields better performance by the mentioned metrics, it is chosen as the best option.

4.2.2.2 The effect of adding a CRF layer

To observe the effectiveness of an added CRF layer on performance, the first and third columns, and the second and fourth columns in Table 4.2 can be compared within themselves. In terms of overall performance, contrary to the initial assumption, the

added CRF layer results in worse performance by all metrics except macro precision and micro precision.

As seen in Table 4.3, in terms of trigger performance, using a CRF layer results in worse average micro F1, as seen in the difference in scores between the first and third columns and second and fourth columns. The same observation can be done for the average micro F1 score for arguments as well.

When the performance is compared on individual triggers and arguments, in the first and third columns, it can be seen that the one without a CRF layer outperforms its counterpart in all triggers except one. In the second and fourth columns, the one without a CRF layer outperforms in all triggers. One difference in terms of performance that strikes the eye is in CWC-change scores. For this argument, there is a wide difference between the model that uses a CRF layer and IOB tags and others. When model predictions are checked it is seen that 69 out of 307 occurrences of this trigger, it spans more than two tokens. However, this model is only able to predict three of these multi-token triggers correctly, whereas others are able to achieve better performance. It seems like the increased number of labels and rare transitions force the model to favor single token predictions for this trigger.

In Person-nationality, the model without CRF layer and IOB tags, the model achieves F1 scores of 40.0, 70.6, 75.0, 69.0 and 80.7 in different folds respectively. Since the splits are made at the document level, argument and trigger ratios between folds do not reflect the document-level ratios. Thus, in the first fold, the model makes its prediction on fewer examples and its performance does not generalize on the specific examples on that fold.

Overall, since not using a CRF layer yields better performance based on the mentioned analysis, not using CRF is selected as the best option.

4.2.2.3 The effect of separating trigger and Argument Extraction

Although the model that does not use IOB labels or a CRF layer provides the best performance among one-stage models, the performance it provides on triggers has room for improvement. Thus a variant of this model that separates trigger and

argument extraction, referred to as the two-stage model, was tested. Basically, the same one-stage model is applied separately on triggers and arguments in the two-stage model. The overall results of this experiment can be seen in Table 4.2. The two-stage model that does not use IOB tags provides the best results across all metrics by a margin.

When the performance at the trigger- and argument-level is compared as seen in Table 4.5, it is seen that the two-stage model performs significantly better at trigger extraction. Not only its average micro F1 score is higher, its error margin is also lower. For all triggers except CIM-action, it outperforms the one-stage model. In the case of CIM-action, the performance between the two models are similar when error margins are considered.

When the two-stage model's performance on arguments is considered, it provides slightly better performance in terms of average micro F1 score. When its performance is compared at the trigger- and argument-level, it is seen that the model outperforms the one-stage variant at seven out of twelve arguments, while attaining similar performance in the ones it does not outperform. For critical arguments like amount, the performance of the best two-stage model is better than not only the one-stage model without CRF layer and IOB tags, but all one-stage models.

Across folds, the two-stage model provides more consistent results as seen in the error margins in Table 4.5. When compared with the error margins in Table 4.4, it seen that the two-stage model provides best results across all models in terms of average micro F1 scores and errors in both triggers and arguments.

The repeated IOB tag comparison also highlights the importance of not using IOB labels once more, as reflected in the results seen in Table 4.4.

4.2.2.4 The selected trigger and argument Extraction Model

As the experiment results conclude, best trigger and argument extraction performance is achieved when triggers and arguments are extracted using two separate models, without IOB tag prediction and without the additional CRF layer. Although running two separate models is costly in terms of both time and computing resources, the

Table 4.2 : Overall performance comparison for one-stage models.

CRF	no	no	yes	yes
IOB	no	yes	no	yes
Macro				
Precision	78.3 ± 1.6	77.7 ± 1.5	**79.5 ± 1.5**	76.4 ± 1.6
Recall	83.8 ± 1.3	**84.5 ± 1.3**	82.3 ± 1.3	83.0 ± 1.5
F1	**80.5 ± 1.3**	**80.5 ± 1.3**	80.2 ± 1.3	78.9 ± 1.5
Micro				
Precision	79.0 ± 1.4	78.5 ± 1.4	**79.3 ± 1.4**	76.9 ± 1.5
Recall	86.1 ± 1.0	**86.8 ± 1.0**	84.9 ± 1.0	86.0 ± 1.1
F1	82.1 ± 1.2	**82.1 ± 1.1**	81.7 ± 1.2	80.8 ± 1.2

significantly improved performance makes it the right choice. The best one-stage model provides a macro F1 score of 80.5 ± 1.3 and micro F1 score of 82.1 ± 1.2, while the best performing two-stage model provides a macro F1 score of 81.5 ± 1.1 and a micro F1 score of 82.5 ± 1.1. Moreover, since triggers are key for event extraction, the better performance of the two-stage model provides for triggers will also result in better performance in the second step of the problem.

Table 4.3 : Comparison of micro F1 scores for one-stage event argument/trigger recognition models.

CRF	no	no	yes	yes
IOB	no	yes	no	yes
Triggers				
NTC-cause	**92.6 ± 0.7**	92.0 ± 1.2	91.1 ± 1.4	90.8 ± 1.0
CC-action	77.4 ± 3.3	**79.8 ± 1.9**	78.4 ± 3.1	73.0 ± 3.8
CWC-change	**72.0 ± 3.6**	71.1 ± 4.2	71.2 ± 1.8	62.4 ± 3.1
CIM-action	**89.9 ± 1.3**	88.6 ± 1.0	89.7 ± 1.4	88.3 ± 0.8
Trig. Avg.	**83.0 ± 2.3**	82.9 ± 2.1	82.6 ± 1.9	78.6 ± 2.2
Arguments				
Title-name	75.7 ± 1.4	**76.7 ± 1.5**	73.7 ± 2.9	71.4 ± 3.1
Person-nationality	**77.2 ± 4.5**	68.9 ± 5.1	74.6 ± 5.6	71.1 ± 6.6
Person-name	83.3 ± 2.0	83.0 ± 2.2	83.2 ± 1.5	**83.8 ± 1.4**
Person- ID number	**77.6 ± 2.3**	74.5 ± 2.5	77.2 ± 1.9	77.3 ± 2.0
NTC - ann. number	**99.6 ± 0.4**	99.5 ± 0.3	97.4 ± 0.3	99.5 ± 0.2
Money - num. shares	64.3 ± 3.8	70.0 ± 3.2	**70.1 ± 2.2**	65.5 ± 5.2
Money - amount	61.5 ± 2.9	**62.9 ± 2.1**	62.5 ± 4.0	61.6 ± 3.1
General - duration	**86.4 ± 1.4**	86.2 ± 0.8	84.9 ± 1.0	86.2 ± 0.7
General - address	86.7 ± 0.9	87.6 ± 1.0	**88.2 ± 0.9**	84.9 ± 1.5
Date - text	**89.6 ± 0.6**	89.2 ± 0.6	87.7 ± 1.8	87.9 ± 1.4
CWC - term	67.6 ± 2.1	**70.2 ± 1.7**	68.1 ± 1.6	68.9 ± 0.8
AuthType - type	89.9 ± 1.6	90.2 ± 1.2	89.9 ± 1.5	**91.2 ± 0.9**
Arg. Avg.	**80.0 ± 2.0**	79.9 ± 1.9	79.8 ± 2.1	79.1 ± 2.2

Table 4.4 : Overall performance comparison for the best one-stage model and the two-stage model.

Stages	one	two
Macro		
Precision	78.3 ± 1.6	**79.3 ± 1.4**
Recall	83.8 ± 1.3	**84.7 ± 1.0**
F1	80.5 ± 1.3	**81.5 ± 1.1**
Micro		
Precision	79.0 ± 1.4	**79.4 ± 1.3**
Recall	86.1 ± 1.0	**86.3 ± 0.9**
F1	82.1 ± 1.2	**82.5 ± 1.1**

Table 4.5 : Comparison of F1 scores for the best one-stage model and the two-stage model.

Stages	one	two
Event	**Argument**	
Triggers		
NTC-cause	92.6 ± 0.7	**92.8 ± 1.2**
CC-action	77.4 ± 3.3	**78.6 ± 1.4**
CWC-change	72.0 ± 3.6	**76.8 ± 1.9**
CIM-action	**89.9 ± 1.3**	89.6 ± 1.0
Trig. Avg.	83.0 ± 2.3	**84.5 ± 1.4**
Arguments		
Title-name	75.7 ± 1.4	**76.6 ± 1.4**
Person-nationality	**77.2 ± 4.5**	76.7 ± 2.3
Person-name	**83.3 ± 2.0**	81.5 ± 1.6
Person- ID number	**77.6 ± 2.3**	76.3 ± 1.7
NTC - ann. number	**99.6 ± 0.4**	99.0 ± 0.5
Money - num. shares	64.3 ± 3.8	**71.7 ± 2.8**
Money - amount	61.5 ± 2.9	**64.2 ± 3.1**
General - duration	**86.4 ± 1.4**	85.6 ± 1.3
General - address	86.7 ± 0.9	**86.8 ± 1.0**
Date - text	89.6 ± 0.6	**89.6 ± 0.5**
CWC - term	67.6 ± 2.1	**69.8 ± 1.4**
AuthType - type	89.9 ± 1.6	**90.8 ± 1.8**
Arg. Avg.	80.0 ± 2.0	**80.7 ± 1.6**

4.3 Event Role Extraction

Event role extraction refers to the assignment of roles between entities and triggers. It determines which triggers and arguments belong to the same event and classifies this relation. To this end, three baselines were provided. The first baseline extracts events using hand-crafted rules. The second one applies the approach in [1], aiming to extract event records by modeling the problem as a table filling problem. Four changes are made to this model to improve the performance and their effects are reported.

4.3.1 Evaluation metrics

Since each event produces an event in a corresponding table, as defined in Section 2, the overlap between the produces event records and the ground truth event record is measured as defined by [1]. Sample event records for CIM, CWC, CC and NTC can be seen in Table 4.6.

Figure 4.1 depicts the event extraction evaluation process. For each document in the dataset, after the model makes its predictions, the populated tables for the document are joined. The same operation is applied for ground truth tables as well. The predicted records are matched with ground truths. For matching, best-matches determined by the maximum number of matching values are considered. After the records are matched, for each role, the number of true positives, false positives and false negatives are calculated. The true positive, false positive and false negative statistics are aggregated at the event- and role-level for all documents. Then, using these statistics, macro and micro precision, recall and F1 scores are calculated.

4.3.2 Rule-based event extraction

As the first event extraction baseline for TRG-EE, a rule-based event extraction model was developed. After triggers and arguments for a document are retrieved, the model goes over the triggers and eligible arguments based on their types. For each event type, a different set of rules is applied. The rules are determined based on manual analysis of the labeled data and patterns extracted from it.

Table 4.6 : Sample event records.

CIM		
action	atanmıştır	assigned as
title - name	Yönetim Kurulu Üyesi	Board Member
title - title holders - name	Ayşe Yılmaz	Ayşe Yılmaz
title - title holders - ID number	123********	123********
title - title holders - address	Beşiktaş/İstanbul	Beşiktaş/İstanbul
title - title holders - nationality	TC Vatandaşı	Turkish Citizen
title - valid from	10.10.2020	10.10.2020
title - valid until	10.10.2022	10.10.2022
title - duration	2 yıllığına	for 2 years
auth. type - type	müştereken	jointly
auth. type - shared with - name	Ali Kaya	Ali Kaya
CWC		
change	artırılarak	by increasing
change amount-amount	10.000.000	10.000.000
new capital-amount	110.000.000	110.000.000
new capital - num. shares	1000000	1000000
new capital - share price	10	10
new capital - shares - amount	200	200
old capital - amount	100.000.000	100.000.000
CC		
action	verilmiştir	given
duration	3 ay süre ile	for 3 months
term	geçici konkordato mühleti	temporary composition
valid from	27.09.2010	27.09.20
NTC		
cause	birleşmeden dolayı	due to merger
ann number	3. İLAN	3RD ANNOUNCEMENT
address	Kadıköy/İstanbul	Kadıköy/İstanbul
duration	en geç 3 ay içerisinde	within 3 months at the latest
valid from	02.12.2020	02.12.2020

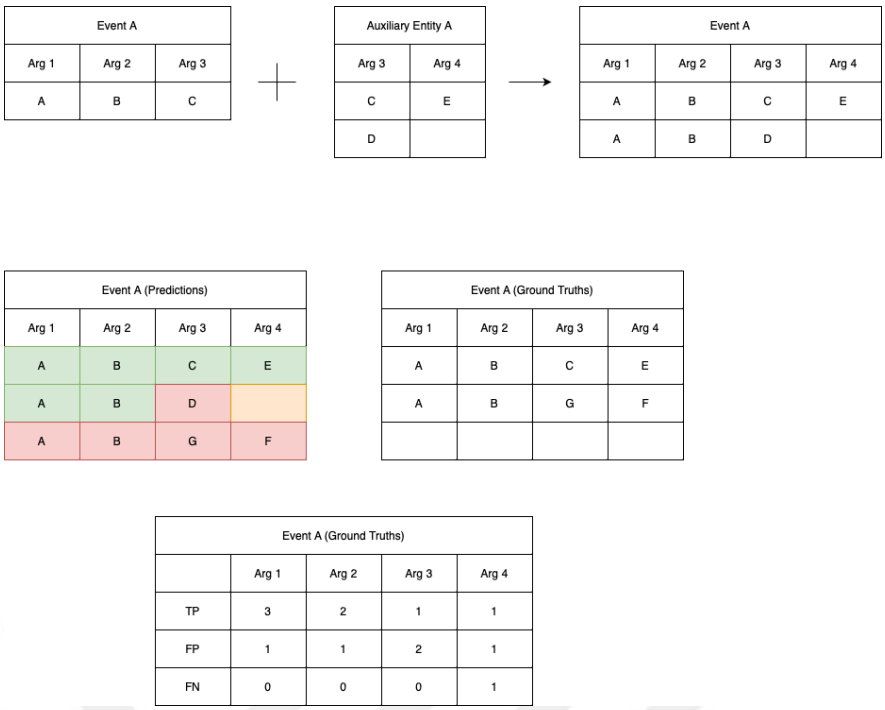

Figure 4.1 : Preprocessing for event extraction evaluation. True positives are colored green, false positives are colored red and false negatives are colored orange.

In Composition with Creditors and Notice to Creditors events, arguments appear in the same sentence as the trigger. Thus, any eligible entity with a suitable type for a given role is directly matched with the trigger.

In Change in Management Entities, auxiliary arguments such as Title, Person and Authorization Type appear. Thus, after the triggers are filtered, if a role requires an auxiliary argument, that entity is extracted separately using its own trigger. In most cases, the trigger and arguments of Title, Person and Authorization Type entities appear in the same sentence. Thus, these are extracted in the same manner as Composition with Creditors or Notice to Creditors events. Then, eligible arguments are assigned to the trigger based on distance, preferring the arguments that appear in the same sentence over others.

Arguments of Change in Working Capital events usually scatter across sentences and they are usually auxiliary entities of Money type. Thus, first, the money entities are resolved. By looking for patterns in the positioning of arguments such as number of shares and money types around it, share price arguments are determined. Then, using the extracted information, these arguments are assigned to one of the amount

arguments either on their left or right to complete the candidate capital arguments for the event. After all money entities are resolved, based on their ordering and distance to the trigger, they are matched with the roles old capital, new capital or change amount.

4.3.3 Doc2EDAG

For the second baseline, Doc2EDAG introduced in [1] was adapted to TRG-EE. As mentioned in Chapter 1, Doc2EDAG treats the problem of event extraction as a table filling task. It tries to predict event records by defining the table as a directed acyclic graph, in which each node corresponds to one column of the table. First, token representations are produced using a transformer model. By applying max-pooling on token representations over argument boundaries and sentences, entity and sentence representations are produced respectively. Then, these represetations are stacked and passed to another transformer, which incorporates document-level information into these representations. Max-pooling is also applied over arguments that have the same surface form to relate the instances of the same argument in different sentences. Since Doc2EDAG converts tables to directed acyclic graphs (DAG), it proposes a path expansion strategy to decode events using DAGs. At each node, a memory tensor is fed to a third transformer. The memory tensor contains candidate arguments and representations of already extracted arguments. At each node, for the given role, this transformer predicts which of the candidates will be assigned to this role. For each predicted candidate, the memory tensor is updated with the assigned argument and the path is expanded for the next node. From start to end, expanded paths give event records.

Since Doc2EDAG was introduced for Chinese, entity extraction part of the model had to be altered. The original entity extraction part works on characters due to Chinese, thus, it was changed to work on tokens. The BERT model finetuned on TRG text was used to generate token embeddings. Since the event schema is completely different, table definitions were adapted to the TRG-EE schema as well. For each event, joined versions of related tables were used. One of the key drawbacks of Doc2EDAG that became apparent in TRG-EE implementation is its infeasible resource use. For example, in documents that contain over 10 instances of the same event, CIM, path

expansion quickly consumes memory when the path can be expanded for all candidates at least during training. Thus, such examples were excluded from the training process. Also, due to differences between Chinese and Turkish, document and sentence lengths change drastically. Thus, maximum sentence length was increased to 256 from 128 tokens and document length was capped at 10 sentences. Due to hardware limitations, batch size per GPU was reduced to 32. The model was trained over five folds for 100 epochs. The best performing epoch was determined by overall micro F1 score over predicted arguments in validation set.

4.3.4 Experiment results

The rule-based model and Doc2EDAG adaptation was compared both in event extraction over gold argument/trigger labels and predicted labels.

Table 4.7 shows the micro F1 scores across folds for experiments. With gold arguments, Doc2EDAG outperforms the rule-based model in all events except CC by a large margin. With predicted labels, Doc2EDAG displays similar performance overall, but it is outperformed by the rule-based model. When gold arguments are used, CC and NTC feature simpler structure and both models achieve fairly good performance. The most challenging of these events is CWC, because it can spread across sentences and feature arguments of the same type in different roles. In this complex event, Doc2EDAG outperforms the rule-based model by a margin in both settings.

The poor performance of Doc2EDAG can be attributed to multi-event examples. As shown in [1], Doc2EDAG's performance worsens when the document contains multiple events. In a document that contains CIM events, usually multiple CIM events are present. Since there are many arguments of the same type, there are cases where Doc2EDAG produces multiple event records for a ground truth event record by duplicating the event and leaving different fields empty.

CWC events span multiple sentences and feature a more complex event structure. The structure requires complex rules to cover all cases, thus the rule-based baseline does not have a good performance to begin with. When predicted inputs are used, the performance worsens even more, since missing arguments distrupt the assumptions

Table 4.7 : Overall performance comparison for event extraction baselines in terms of micro F1 score.

Model Input Type	Rule-based gold	Doc2EDAG gold	Rule-based pred	Doc2EDAG pred
Event				
CIM	43.1 ± 2.6	**49.1 ± 4.8**	**49.7 ± 2.0**	33.4 ± 3.4
CWC	24.3 ± 0.9	**52.8 ± 10.7**	29.7 ± 1.0	**43.5 ± 3.7**
CC	**98.8 ± 0.1**	94.2 ± 1.5	**68.9 ± 0.9**	62.5 ± 6.8
NTC	99.1 ± 0.2	**99.4 ± 0.3**	72.2 ± 1.1	**77.4 ± 8.8**
Average	66.3 ± 1.0	**73.9 ± 4.3**	**55.1 ± 1.3**	54.2 ± 5.6

of the model. Doc2EDAG on the other hand, is able to adapt and provides better performance compared to the rule-based model, while performing worse on predicted labels. Unlike CIM events, multiple CWC events usually do not appear together. Thus, the model performs better in CWC events compared to CIM events despite being more complex with multiple arguments of the same type.

Role-level comparison with gold arguments can be seen in Table 4.8. The rule-based model outperforms Doc2EDAG in simpler roles. In the case of CIM events, the ones the rule-based model performs well on usually appear in the same sentence as the other arguments of the events they belong to, like action, title-name, title-address etc.. On the other hand, like roles authorization type-type not only are rarer, but also appear in various patterns. Thus, Doc2EDAG is better at capturing such roles. Since one action trigger can create multiple event records, Doc2EDAG tends to use the same trigger in multiple roles, causing mismatches and poorer performance. When predictions are examined, it is seen that Doc2EDAG can produce multiple records with missing fields for a gold event record. This is especially apparent in CIM events, where multiple fields can be left empty in actual cases.

In CWC, Doc2EDAG outperforms the rule-based model in all roles. As mentioned earlier, this type of event is tricky because of how its arguments spread and how arguments of the same type appear in different roles. Some arguments are in relation with other arguments, like shares and share price. Such intricate relations are hard to capture by rules and Doc2EDAG's superior performance to the rule-based model in harder roles like new capital-share price or new capital-shares-amount reflect this. Thanks to the way Doc2EDAG assigns roles to spans and its document context

modeling approach, the model is able to distinguish roles, even in the case of new-capital-shares-amount. The rule-based model fails at filling this role in the event altogether, while Doc2EDAG achieves a good performance.

In CC, the rule-based model benefits from the simpler structure and outperforms Doc2EDAG and Doc2EDAG outperforms the rule-based model in NTC. Both models achieve F1 scores over 90 in these event types, with almost perfect peformance in NTC.

Similar observations can be done on the results on predicted inputs in Table 4.9. In CIM, Doc2EDAG performs better than the rule-based model in only four roles. It must be mnetioned that the rule-based model uses the two-stage model selected in the trigger and argument extraction experiments, while Doc2EDAG trains the BERT model used to finetune the two-stage model from scratch to predict span boundaries to be used in roles. When model predictions are examined, it is seen that Doc2EDAG's role filling performance heavily depends on the performance of the span extraction phase. Thus, its performance suffers due to failing to extract valid spans, especially in CIM.

In CWC, Doc2EDAG is able to outperform the rule-based model in all roles. When model predictions are examined, it is seen that both models work with partial span predictions, but Doc2EDAG benefits from its superior path expansion approach as opposed to the rule-based model's static rule set.

In CC, the rule-based model outperforms Doc2EDAG, while Doc2EDAG outperforms it in NTC. In NTC, the rule-based model fails in locating durations and dates for valid from role while Doc2EDAG predicts these spans correctly.

Although prediction performance of Doc2EDAG worsens on predicted arguments as seen in Table 4.7 and Table 4.9, when the model predictions are examined, it is seen that the events it is able to extract tend to be more correct. Since the model uses its own predictions for arguments, it seems more confident at understanding which argument belongs to which field of which event record. However, in cases where it cannot extract an event correctly, it usually is not able to fill majority of the fields, resulting in the performance shown. In more complex events, this can be to its benefit as seen in Table Table 4.7.

Table 4.8 : Comparison of micro F1 scores for the rule-based event extraction model and Doc2EDAG adaptation with gold arguments on role-level.

	Model	Rule-based	Doc2EDAG	
	Input Type	gold	gold	Support
Event	Role			
CIM	action	51.1 ± 2.6	**55.0 ± 3.9**	1175
CIM	auth. type-shared with-name	0.0 ± 0.0	**19.9 ± 5.1**	226
CIM	auth. type-type	18.5 ± 3.8	**68.3 ± 3.6**	397
CIM	title-duration	60.6 ± 3.9	**65.3 ± 4.8**	374
CIM	title-name	48.4 ± 2.6	**51.6 ± 4.8**	1050
CIM	title-title holders-address	39.6 ± 2.8	**45.0 ± 5.5**	884
CIM	title-title holders-ID number	**45.8 ± 2.7**	39.7 ± 6.8	755
CIM	title-title holders-name	44.0 ± 2.2	**46.1 ± 4.2**	1094
CIM	title-title holders-nationality	**31.6 ± 3.8**	26.5 ± 5.3	215
CIM	title-valid from	31.4 ± 1.9	**47.8 ± 11.2**	33
CIM	title-valid until	**23.1 ± 2.7**	19.6 ± 4.3	398
CIM	Average	35.8 ± 2.6	**44.1 ± 5.4**	
CWC	change	51.1 ± 1.3	**55.8 ± 10.3**	384
CWC	change amount	40.1 ± 2.3	**50.0 ± 10.3**	296
CWC	new capital-amount	12.3 ± 2.0	**50.0 ± 8.3**	409
CWC	new capital-num. shares	33.2 ± 2.0	**50.7 ± 9.7**	301
CWC	new capital-share price	0.0 ± 0.0	**48.7 ± 10.1**	300
CWC	new capital-shares-amount	0.0 ± 0.0	**49.1 ± 10.7**	254
CWC	old capital-amount	9.1 ± 2.1	**55.8 ± 7.1**	230
CWC	Average	20.8 ± 1.4	**51.5 ± 9.5**	
CC	action	**99.3 ± 0.0**	93.3 ± 0.0	417
CC	duration	**99.1 ± 0.2**	94.1 ± 1.8	352
CC	term	**98.8 ± 0.2**	94.4 ± 1.4	415
CC	valid from	**97.6 ± 0.1**	94.8 ± 1.6	302
CC	Average	**98.7 ± 0.2**	94.1 ± 1.6	
NTC	address	97.7 ± 0.7	**99.4 ± 0.4**	493
NTC	ann. number	**100.0 ± 0.0**	99.4 ± 0.5	494
NTC	cause	**100.0 ± 0.0**	**100.0 ± 0.0**	494
NTC	duration	**98.7 ± 0.2**	98.6 ± 1.0	492
NTC	valid from	98.7 ± 0.2	**99.7 ± 0.1**	481
NTC	Average	99.0 ± 0.2	**99.4 ± 0.4**	
Overall		53.0 ± 1.5	**63.6 ± 5.0**	

As seen in Table 4.7, both models end up with reduced performance when predicted arguments are used. Moreover, when role-level performance is compared, as seen in Table 4.8 and Table 4.9, it is seen that the rule-based model has poor performance on more complex fields. Overall, the rule-based model is able to handle simpler events and roles with simpler patterns, but it is brittle.

For Doc2EDAG, a couple issues stand out. The most important of these issues is predicting path expansion without considering suitable argument types. This results in the model producing completely invalid events. Also, during the training process, the model unnecessarily learns to make negative predictions on incompatible arguments. Type-checking can be handy in reducing invalid event records produced by the model. The observations also show that the model's event extraction performance is greatly affected by its argument prediction performance. Since the model learns argument extraction and path expansion jointly, there exists a trade-off during training. The model predicts roles on token spans directly, thus for arguments of the same type that appear in different roles, this can be non-ideal. Although the authors show that the path expansion memory increases performance as shown in [1], it is a costly component. The added transformer results in three times the training time required and passing around large matrices also increases the RAM requirements. This, in turn, requires reduced sentence length and counts while modeling the document context.

4.3.5 Proposed changes

To improve the performance of Doc2EDAG, four changes are proposed and their effects are observed separately with experiments. Doc2EDAG is a large model that takes a lot of time to train. Despite the smaller size of the dataset, with two GPUs, one fold of experiments takes around 7.5 hours. The slow performance can be attributed to the calculations done during path expansion, which increases the training time two times. As the first change, the effect of removing path expansion memory is observed.

Table 4.10 shows event-level performance comparison when path expansion memory is turned off. Regardless of whether gold arguments or predicted arguments are used for event extraction, the overall performance of the model is affected severely when path expansion memory is turned off. This supports the experiments in [1]. As seen

Table 4.9 : Comparison of F1 scores for the rule-based event extraction model and Doc2EDAG adaptation with predicted arguments.

	Model	Rule-based	Doc2EDAG	
	Input Type	pred	pred	Support
Event	Role			
CIM	action	63.1 ± 2.4	**41.6 ± 2.0**	1175
CIM	auth. type-shared with-name	0.0 ± 0.0	**8.7 ± 3.2**	226
CIM	auth. type-type	8.9 ± 2.4	**52.7 ± 4.4**	397
CIM	title-duration	13.8 ± 2.1	**44.4 ± 4.6**	374
CIM	title-name	**61.6 ± 2.4**	38.4 ± 3.3	1050
CIM	title-title holders-address	**57.5 ± 2.8**	28.1 ± 2.8	884
CIM	title-title holders-ID number	**38.8 ± 2.5**	22.2 ± 9.3	755
CIM	title-title holders-name	**51.9 ± 1.9**	31.7 ± 2.9	1094
CIM	title-title holders-nationality	**43.8 ± 6.5**	22.6 ± 4.6	215
CIM	title-valid from	**3.0 ± 0.7**	0.0 ± 0.0	33
CIM	title-valid until	**58.7 ± 2.1**	4.3 ± 2.4	398
CIM	Average	**36.5 ± 2.3**	26.8 ± 3.6	
CWC	change	53.9 ± 2.0	**55.7 ± 4.5**	384
CWC	change amount	31.3 ± 1.9	**37.3 ± 6.9**	296
CWC	new capital-amount	25.8 ± 2.9	**33.9 ± 2.3**	409
CWC	new capital-num. shares	34.1 ± 1.2	**44.0 ± 3.2**	301
CWC	new capital-share price	32.4 ± 1.8	**45.2 ± 3.6**	300
CWC	new capital-shares-amount	0.0 ± 0.0	**42.2 ± 4.9**	254
CWC	old capital-amount	8.0 ± 1.3	**37.3 ± 2.9**	230
CWC	Average	26.5 ± 1.6	**42.2 ± 4.0**	
CC	action	**78.4 ± 1.3**	74.2 ± 3.6	417
CC	duration	46.5 ± 1.1	**48.0 ± 12.0**	352
CC	term	**71.2 ± 1.2**	62.9 ± 4.8	415
CC	valid from	**71.1 ± 1.0**	42.2 ± 17.3	302
CC	Average	**66.8 ± 1.1**	56.8 ± 9.4	
NTC	address	**90.2 ± 1.9**	58.4 ± 18.8	493
NTC	ann. number	**96.8 ± 0.6**	70.8 ± 14.6	494
NTC	cause	92.6 ± 1.3	**92.6 ± 1.1**	494
NTC	duration	14.8 ± 1.2	**90.8 ± 1.7**	492
NTC	valid from	34.8 ± 1.5	**61.2 ± 17.3**	481
NTC	Average	65.8 ± 1.3	**74.8 ± 10.7**	
Overall		43.8 ± 1.8	**44.1 ± 5.9**	

Table 4.10 : Overall performance comparison when path expansion memory is turned off in terms of micro F1 score.

Model	Doc2EDAG	Doc2EDAG - Path Expansion Memory	Doc2EDAG	Doc2EDAG - Path Expansion Memory
Input Type	gold	gold	pred	pred
Event				
CIM	**49.1 ± 4.8**	28.6 ± 1.8	**33.4 ± 3.4**	18.2 ± 4.9
CWC	52.8 ± 10.7	**54.0 ± 3.6**	**43.5 ± 3.7**	31.5 ± 7.6
CC	**94.2 ± 1.5**	80.4 ± 1.9	**62.5 ± 6.8**	54.1 ± 13.8
NTC	99.4 ± 0.3	**99.7 ± 0.1**	**77.4 ± 8.8**	75.9 ± 19.0
Average	**73.9 ± 4.3**	65.7 ± 1.8	**54.2 ± 5.6**	44.9 ± 11.3

in the table, when gold arguments are used, the performance improves in CWC and NTC, while the improvements in NTC are minimal. However, when predicted labels are used, the performance is reduced for all events. Path expansion memory seems to be effective when extracting CIM events especially, since multiple events of this type usually appear in the same announcement. When partial information is present, path expansion memory proves to be an important mechanism for event extraction despite its added time and computation cost.

Table 4.11 shows role-level performance comparison when path expansion memory is turned off. In line with the event-level observations listed above, path expansion hinders performance for all roles in CIM and CC. NTC, on the other hand, being the simplest event that usually appears once in a document with usually all arguments present, seems to be the least affected and the performance for all roles except one is improved. In CWC, turning off improves the performance for most roles, albeit slightly in most cases.

Table 4.12 shows performance comparison with predicted arguments when path expansion memory is turned off. Under predicted labels, the model without path expansion memory performs worse for all roles in more complex events like CIM and CWC. Although the model is able to outperform the variant with path memory in some roles in simpler events like CC and NTC, the overall performance is reduced in all events. Since performance with predicted labels would be more indicative of

Table 4.11 : Comparison of F1 scores with gold arguments when path expansion memory is turned off.

	Model	Doc2EDAG	Doc2EDAG - Path Expansion Memory	
	Input Type	gold	gold	Support
Event	Role			
CIM	action	**55.0 ± 3.9**	46.3 ± 2.4	1175
CIM	auth. type-shared with-name	**19.9 ± 5.1**	2.4 ± 1.4	226
CIM	auth. type-type	**68.3 ± 3.6**	39.9 ± 3.1	397
CIM	title-duration	**65.3 ± 4.8**	30.5 ± 3.6	374
CIM	title-name	**51.6 ± 4.8**	39.3 ± 1.8	1050
CIM	title-title holders-address	**45.0 ± 5.5**	21.6 ± 1.4	884
CIM	title-title holders-ID number	**39.7 ± 6.8**	10.6 ± 1.5	755
CIM	title-title holders-name	**46.1 ± 4.2**	16.4 ± 2.9	1094
CIM	title-title holders-nationality	**26.5 ± 5.3**	23.7 ± 4.1	215
CIM	title-valid from	**47.8 ± 11.2**	26.3 ± 7.7	33
CIM	title-valid until	**19.6 ± 4.3**	9.9 ± 2.8	398
CIM	Average	**44.1 ± 5.4**	24.3 ± 3.0	
CWC	change	55.8 ± 10.3	**59.0 ± 3.9**	384
CWC	change amount	50.0 ± 10.3	**55.7 ± 4.1**	296
CWC	new capital-amount	**50.0 ± 8.3**	45.9 ± 2.1	409
CWC	new capital-num. shares	50.7 ± 9.7	**55.8 ± 3.9**	301
CWC	new capital-share price	48.7 ± 10.1	**51.8 ± 3.8**	300
CWC	new capital-shares-amount	**49.1 ± 10.7**	48.9 ± 5.6	254
CWC	old capital-amount	55.8 ± 7.1	**57.0 ± 4.4**	230
CWC	Average	51.5 ± 9.5	**53.4 ± 4.0**	
CC	action	**93.3 ± 0.0**	84.4 ± 2.8	417
CC	duration	**94.1 ± 1.8**	77.6 ± 1.8	352
CC	term	**94.4 ± 1.4**	73.6 ± 0.7	415
CC	valid from	**94.8 ± 1.6**	84.7 ± 2.2	302
CC	Average	**94.1 ± 1.6**	80.1 ± 1.9	
NTC	address	99.4 ± 0.4	**99.9 ± 0.1**	493
NTC	ann. number	99.4 ± 0.5	**99.9 ± 0.1**	494
NTC	cause	**100.0 ± 0.0**	99.8 ± 0.1	494
NTC	duration	98.6 ± 1.0	**99.4 ± 0.3**	492
NTC	valid from	**99.7 ± 0.1**	**99.7 ± 0.1**	481
NTC	Average	99.4 ± 0.4	**99.7 ± 0.1**	
Overall		**63.6 ± 5.0**	54.1 ± 2.5	

the performance in real world, despite time and computational cost, path expansion memory should be kept on.

Since the removal of the CRF layer was observed to affect event argument extraction performance positively, the same approach was applied to Doc2EDAG as the second change.

Table 4.13 shows the event-level performance comparison when the CRF layer is removed from Doc2EDAG. As seen in the table, under gold arguments, the performance improves in all events and under predicted arguments, it improves in all events except CWC. With gold arguments, the overall performance improvements is around 1 point while the gap widens to around 6 points with predicted arguments. These results are reflect the results in event argument extraction experiments.

Table 4.14 shows role-level performance comparison with gold arguments when the CRF layer is removed. The performance of simpler events like CC and NTC are not affected much, but the role level performance in CIM and CWC seems to be improved greatly. In CIM event, the performance seems to be improved in roles related to the auxiliary Person entity while the opposite seems to be the case for the auxiliary Authorization Type entity. The model performance greatly improves for commonly found time-related roles like Title-valid until. In CWC, al roles except one observe improvements around 2 points while new capital-shares-amount observes a decline of around 2 points.

Table 4.15 shows role-level performance comparision with predicted arguemnts when CRF layer is removed. The performance in NTC seems to be increased drastically for all roles. In CC, the performance seems to be more stable across all roles. When the output of the model is manually checked, it is observed that in cases where the original model fails, the variant model seems to be able to extract the arguments missed by the original correctly or match extracted arguments to correct event records. In CWC, some roles observe slight improvements while other observe slight declines. In CIM, performance for almost all fields that correspond to arguments of Person entity and title-valid until observe improvements just like in the case of gold arguments. However, unlike the observations when gold arguments are used, fields related to

Table 4.12 : Comparison of F1 scores with predicted arguments when path expansion memory is turned off.

	Model	Doc2EDAG	Doc2EDAG - Path Expansion Memory	
	Input Type	pred	pred	Support
Event	Role			
CIM	action	**41.6 ± 2.0**	22.8 ± 7.4	1175
CIM	auth. type-shared with-name	**8.7 ± 3.2**	1.4 ± 1.3	226
CIM	auth. type-type	**52.7 ± 4.4**	26.9 ± 4.7	397
CIM	title-duration	**44.4 ± 4.6**	21.0 ± 5.2	374
CIM	title-name	**38.4 ± 3.3**	26.5 ± 7.3	1050
CIM	title-title holders-address	**28.1 ± 2.8**	11.9 ± 3.7	884
CIM	title-title holders-ID number	**22.2 ± 9.3**	6.8 ± 2.1	755
CIM	title-title holders-name	**31.7 ± 2.9**	10.2 ± 3.1	1094
CIM	title-title holders-nationality	**22.6 ± 4.6**	17.2 ± 4.3	215
CIM	title-valid from	**0.0 ± 0.0**	**0.0 ± 0.0**	33
CIM	title-valid until	**4.3 ± 2.4**	3.3 ± 1.2	398
CIM	Average	**26.8 ± 3.6**	13.5 ± 3.7	
CWC	change	**55.7 ± 4.5**	40.7 ± 8.7	384
CWC	change amount	**37.3 ± 6.9**	30.0 ± 9.4	296
CWC	new capital-amount	**33.9 ± 2.3**	20.8 ± 4.8	409
CWC	new capital-num. shares	**44.0 ± 3.2**	34.0 ± 7.7	301
CWC	new capital-share price	**45.2 ± 3.6**	29.6 ± 7.2	300
CWC	new capital-shares-amount	**42.2 ± 4.9**	25.8 ± 4.5	254
CWC	old capital-amount	**37.3 ± 2.9**	29.3 ± 7.6	230
CWC	Average	**42.2 ± 4.0**	30.0 ± 7.1	
CC	action	**74.2 ± 3.6**	61.6 ± 13.8	417
CC	duration	48.0 ± 12.0	**50.3 ± 10.3**	352
CC	term	**62.9 ± 4.8**	47.2 ± 10.4	415
CC	valid from	42.2 ± 17.3	**59.4 ± 12.4**	302
CC	Average	**56.8 ± 9.4**	54.7 ± 11.8	
NTC	address	58.4 ± 18.8	**74.7 ± 17.5**	493
NTC	ann. number	70.8 ± 14.6	**77.4 ± 18.1**	494
NTC	cause	**92.6 ± 1.1**	75.5 ± 17.6	494
NTC	duration	**90.8 ± 1.7**	72.3 ± 17.0	492
NTC	valid from	61.2 ± 17.3	**73.3 ± 17.0**	481
NTC	Average	**74.8 ± 10.7**	74.6 ± 17.4	
Overall		**44.1 ± 5.9**	35.2 ± 8.3	

Table 4.13 : Overall performance comparison when the CRF layer is removed in terms of micro F1 score.

Model	Doc2EDAG	Doc2EDAG - CRF	Doc2EDAG	Doc2EDAG - CRF
Input Type	gold	gold	pred	pred
Event				
CIM	49.1 ± 4.8	**50.0 ± 3.0**	33.4 ± 3.4	**34.9 ± 2.6**
CWC	52.8 ± 10.7	**55.2 ± 13.1**	**43.5 ± 3.7**	42.5 ± 4.3
CC	94.2 ± 1.5	**95.1 ± 1.4**	62.5 ± 6.8	**71.6 ± 2.6**
NTC	99.4 ± 0.3	**99.4 ± 0.2**	77.4 ± 8.8	**93.0 ± 0.3**
Average	73.9 ± 4.3	**74.9 ± 4.4**	54.2 ± 5.6	**60.5 ± 2.4**

Authorization Type entity observe an improvement under extraction with predicted arguments. In CWC, some roles observe slight improvements while others observe slight declines and the overall performance for this event is also slightly reduced. With increased role-level performance in three events out of four, the overall performance also increases by 6.8 points, thus removing the CRF layer proves to be an effective improvement.

As the third change transfer learning from event argument extraction models is proposed. Since Doc2EDAG uses a BERT model to produce argument representations, the hypothesis is that replacing this BERT with a BERT model finetuned on the event argument extraction task should improve performance. To this end, the BERT model finetuned on argument extraction in the best performing two-stage model from argument extraction experiments is transferred to Doc2EDAG.

Table 4.16 shows the overall performance comparison when transfer learning is applied. When gold arguments are used, the performance improves in two events while the performance improves in all events when predicted arguments are used. Overall performance improves by 1.3 points and 8.7 points when gold arguments are used and when predicted arguments are used respectively. The model is able to retain its performance between gold arguments and predicted better with transfer learning, especially in the case of NTC.

Table 4.17 shows role-level performance with gold arguments when transfer learning is applied. More CC roles observe improvements with transfer learning while more NTC roles observe improvements without. However, NTC achieves almost perfect

Table 4.14 : Comparison of F1 scores with gold arguments when CRF layer is removed.

	Model	Doc2EDAG	Doc2EDAG - CRF	
	Input Type	gold	gold	Support
Event	Role			
CIM	action	**55.0 ± 3.9**	54.3 ± 2.5	1175
CIM	auth. type-shared with-name	**19.9 ± 5.1**	13.6 ± 5.0	226
CIM	auth. type-type	**68.3 ± 3.6**	61.0 ± 2.1	397
CIM	title-duration	**65.3 ± 4.8**	64.5 ± 2.8	374
CIM	title-name	**51.6 ± 4.8**	51.0 ± 5.3	1050
CIM	title-title holders-address	45.0 ± 5.5	**46.9 ± 2.8**	884
CIM	title-title holders-ID number	39.7 ± 6.8	**48.4 ± 4.0**	755
CIM	title-title holders-name	46.1 ± 4.2	**47.1 ± 1.9**	1094
CIM	title-title holders-nationality	26.5 ± 5.3	**43.2 ± 3.9**	215
CIM	title-valid from	**47.8 ± 11.2**	44.3 ± 10.3	33
CIM	title-valid until	19.6 ± 4.3	**38.1 ± 3.8**	398
CIM	Average	44.1 ± 5.4	**46.6 ± 4.0**	
CWC	change	55.8 ± 10.3	**56.9 ± 12.1**	384
CWC	change amount	50.0 ± 10.3	**53.4 ± 12.0**	296
CWC	new capital-amount	50.0 ± 8.3	**52.9 ± 11.2**	409
CWC	new capital-num. shares	50.7 ± 9.7	**52.8 ± 11.5**	301
CWC	new capital-share price	48.7 ± 10.1	**50.6 ± 12.6**	300
CWC	new capital-shares-amount	**49.1 ± 10.7**	47.7 ± 12.0	254
CWC	old capital-amount	55.8 ± 7.1	**67.0 ± 7.2**	230
CWC	Average	51.5 ± 9.5	**54.5 ± 11.2**	
CC	action	93.3 ± 0.0	**96.2 ± 1.4**	417
CC	duration	94.1 ± 1.8	**94.3 ± 2.3**	352
CC	term	**94.4 ± 1.4**	93.8 ± 1.2	415
CC	valid from	94.8 ± 1.6	**95.4 ± 0.7**	302
CC	Average	94.1 ± 1.6	**94.9 ± 1.4**	
NTC	address	99.4 ± 0.4	**99.7 ± 0.2**	493
NTC	ann. number	99.4 ± 0.5	**100.0 ± 0.0**	494
NTC	cause	**100.0 ± 0.0**	99.6 ± 0.4	494
NTC	duration	**98.6 ± 1.0**	98.4 ± 0.6	492
NTC	valid from	**99.7 ± 0.1**	99.4 ± 0.3	481
NTC	Average	99.4 ± 0.4	**99.4 ± 0.3**	
Overall		63.6 ± 5.0	**65.6 ± 4.8**	

Table 4.15 : Comparison of F1 scores with predicted arguments when CRF layer is removed.

	Model	Doc2EDAG	Doc2EDAG - CRF	
	Input Type	pred	pred	Support
Event	Role			
CIM	action	**41.6 ± 2.0**	38.6 ± 3.1	1175
CIM	auth. type-shared with-name	8.7 ± 3.2	**14.5 ± 8.3**	226
CIM	auth. type-type	52.7 ± 4.4	**53.4 ± 4.8**	397
CIM	title-duration	**44.4 ± 4.6**	42.4 ± 2.2	374
CIM	title-name	38.4 ± 3.3	**38.6 ± 3.1**	1050
CIM	title-title holders-address	28.1 ± 2.8	**31.2 ± 1.5**	884
CIM	title-title holders-ID number	22.2 ± 9.3	**34.1 ± 3.2**	755
CIM	title-title holders-name	**31.7 ± 2.9**	30.6 ± 2.6	1094
CIM	title-title holders-nationality	22.6 ± 4.6	**23.5 ± 3.6**	215
CIM	title-valid from	0.0 ± 0.0	0.0 ± 0.0	33
CIM	title-valid until	4.3 ± 2.4	**23.2 ± 4.7**	398
CIM	Average	26.8 ± 3.6	**30.0 ± 3.4**	
CWC	change	**55.7 ± 4.5**	54.3 ± 4.7	384
CWC	change amount	**37.3 ± 6.9**	34.1 ± 4.8	296
CWC	new capital-amount	33.9 ± 2.3	**35.1 ± 2.7**	409
CWC	new capital-num. shares	44.0 ± 3.2	**45.0 ± 3.9**	301
CWC	new capital-share price	45.2 ± 3.6	**45.7 ± 4.6**	300
CWC	new capital-shares-amount	**42.2 ± 4.9**	35.8 ± 6.7	254
CWC	old capital-amount	37.3 ± 2.9	**40.2 ± 4.7**	230
CWC	Average	**42.2 ± 4.0**	41.5 ± 4.6	
CC	action	**74.2 ± 3.6**	71.4 ± 3.0	417
CC	duration	48.0 ± 12.0	**73.8 ± 2.0**	352
CC	term	62.9 ± 4.8	**67.2 ± 2.1**	415
CC	valid from	42.2 ± 17.3	**76.3 ± 3.1**	302
CC	Average	56.8 ± 9.4	**72.2 ± 2.6**	
NTC	address	58.4 ± 18.8	**92.6 ± 0.4**	493
NTC	ann. number	70.8 ± 14.6	**97.7 ± 0.6**	494
NTC	cause	92.6 ± 1.1	**92.6 ± 0.8**	494
NTC	duration	90.8 ± 1.7	**91.9 ± 1.0**	492
NTC	valid from	61.2 ± 17.3	**90.2 ± 1.2**	481
NTC	Average	74.8 ± 10.7	**93.0 ± 0.8**	
Overall		44.1 ± 5.9	50.9 ± 3.1	

Table 4.16 : Overall performance comparison when transfer learning is applied in terms of micro F1 score.

Model	Doc2EDAG	Doc2EDAG + Transfer Learning	Doc2EDAG	Doc2EDAG + Transfer Learning
Input Type	gold	gold	pred	pred
Event				
CIM	**49.1 ± 4.8**	46.8 ± 3.3	33.4 ± 3.4	**38.3 ± 3.0**
CWC	52.8 ± 10.7	**60.6 ± 10.3**	43.5 ± 3.7	**44.7 ± 5.3**
CC	94.2 ± 1.5	**94.4 ± 1.2**	62.5 ± 6.8	**76.2 ± 3.6**
NTC	**99.4 ± 0.3**	98.9 ± 0.4	77.4 ± 8.8	**92.6 ± 1.4**
Average	73.9 ± 4.3	**75.2 ± 3.8**	54.2 ± 5.6	**62.9 ± 3.3**

scores in both cases and CC's performance is above 94. Transfer learning improves performance across all CWC roles and improves the overall performance by 8.7 points. Interestingly, the opposite is the case for CIM. In all roles except three, Doc2EDAG has better performance. Two of these three roles correspond to the auxiliary Person entity and the model seems to handle ID numbers and nationalities better with transfer learning.

Table 4.18 shows role-level performance with predicted arguments when transfer learning is applied. Under partial information, the model is able to outperform Doc2EDAG in all roles except four. Transfer learning seems to be effective in this setting. In CIM, the performance of all roles that correspond to the auxiliary Person entity seems to improve drastically. The two roles that have a decreased performance seems to be reduced by around 1 point. The overall performance of the model increases by 5.2 points for this event type. As it can be observed in CWC, transfer learning seems to help with numerical expressions. All numerical expressions except one seems to have an improved performance, some with a margin of up to 4.3 points. In CC and NTC events, transfer learning helps the model retain its performance better. Almost all roles in NTC has a performance above 90 points and the average performance is improved by 17.4 points. While CC roles observe a decline 17.6 compared to the performance under gold arguments, compared to Doc2EDAG, the model with transfer learning seems to improve the performance by 20 points in this event type. When model predictions are manually checked, it is observed that not only transfer learning helps the model find more correct argument candidates, their boundaries are also more

Table 4.17 : Comparison of F1 scores with gold arguments when transfer learning is applied.

	Model	Doc2EDAG	Doc2EDAG + Transfer Learning	
	Input Type	gold	gold	Support
Event	Role			
CIM	action	**55.0 ± 3.9**	49.8 ± 3.9	1175
CIM	auth. type-shared with-name	**19.9 ± 5.1**	15.4 ± 4.6	226
CIM	auth. type-type	**68.3 ± 3.6**	58.2 ± 6.5	397
CIM	title-duration	**65.3 ± 4.8**	59.2 ± 4.5	374
CIM	title-name	**51.6 ± 4.8**	47.5 ± 4.3	1050
CIM	title-title holders-address	**45.0 ± 5.5**	43.8 ± 3.3	884
CIM	title-title holders-ID number	39.7 ± 6.8	**46.5 ± 2.8**	755
CIM	title-title holders-name	**46.1 ± 4.2**	44.6 ± 2.6	1094
CIM	title-title holders-nationality	26.5 ± 5.3	**36.8 ± 4.5**	215
CIM	title-valid from	**47.8 ± 11.2**	16.8 ± 4.6	33
CIM	title-valid until	19.6 ± 4.3	**33.7 ± 2.9**	398
CIM	Average	**44.1 ± 5.4**	41.1 ± 4.0	
CWC	change	55.8 ± 10.3	**61.8 ± 8.7**	384
CWC	change amount	50.0 ± 10.3	**58.2 ± 9.7**	296
CWC	new capital-amount	50.0 ± 8.3	**56.6 ± 8.9**	409
CWC	new capital-num. shares	50.7 ± 9.7	**60.1 ± 9.3**	301
CWC	new capital-share price	48.7 ± 10.1	**56.9 ± 10.8**	300
CWC	new capital-shares-amount	49.1 ± 10.7	**55.8 ± 10.5**	254
CWC	old capital-amount	55.8 ± 7.1	**71.9 ± 3.6**	230
CWC	Average	51.5 ± 9.5	**60.2 ± 8.8**	
CC	action	93.3 ± 0.0	**94.0 ± 1.4**	417
CC	duration	94.1 ± 1.8	**95.8 ± 1.0**	352
CC	term	**94.4 ± 1.4**	92.8 ± 0.9	415
CC	valid from	94.8 ± 1.6	**95.2 ± 1.7**	302
CC	Average	94.1 ± 1.6	**94.4 ± 1.2**	
NTC	address	**99.4 ± 0.4**	97.2 ± 2.2	493
NTC	ann. number	99.4 ± 0.5	**100.0 ± 0.0**	494
NTC	cause	**100.0 ± 0.0**	98.7 ± 0.6	494
NTC	duration	98.6 ± 1.0	**98.9 ± 0.5**	492
NTC	valid from	**99.7 ± 0.1**	99.1 ± 0.5	481
NTC	Average	**99.4 ± 0.4**	98.8 ± 0.8	
Overall		63.6 ± 5.0	**64.6 ± 4.3**	

correct, and thus the model matches them with correct fields in correct event records more easily. The role-level performance of the overall model is increased by 9.1 points, which concludes that transfer learning is an viable improvement.

Lastly, since Doc2EDAG does not filter irrelevant argument candidates during path expansion, it considers unsuitable candidates while matching arguments with fields. Assuming that filtering such arguments out during path expansion, a field-aware path expansion mechanism where for each field, only arguments with the matching type are considered for decision. Training with field-aware path expansion also decreases time and computation costs of the model.

Table 4.19 shows the role-level performance comparison when field-aware path expansion is applied. As seen in the table, when gold arguments are used, the performance of the model only improves in CWC and the overall performance is reduced by 3.7 points. When predicted arguments are used, the model is able to outperform Doc2EDAG in CC and NTC by a large margin, however, in more complex events like CIM and CWC, the opposite can be observed. Although the overall performance under predicted arguments improve with field-aware path expansion, the reduction in performance in more complex events are important to highlight.

Table 4.19 shows role-level performance with gold arguments when field-aware path expansion is applied. In all roles except five, field-aware path expansion seems to decrease performance. In NTC, the model is able to attain almost perfect performance both in most roles and overall. In CC, only the performance under action improves. In CWC, change observes the highest improvements while other roles observe a decrease of around 5 points. In CIM, all roles observe decreases, with some up to 20 points. Since NTC is the least complex event with usually only a single argument candidate for each role, it is the least affected. In CWC and CIM, change and action also usually are the only candidates for a field. However, since multiple CIM events appear together, selecting between events of the same type seems to be harder for the model.

Table 4.20 shows role-level performance with gold arguments when field-aware path expansion is applied. The change seems to improve performance under only simpler events like CC and NTC. Since filtering usually leaves a single option for the model

Table 4.18 : Comparison of F1 scores with predicted arguments when transfer learning is applied.

	Model	Doc2EDAG	Doc2EDAG + Transfer Learning	
	Input Type	pred	pred	Support
Event	Role			
CIM	action	41.6 ± 2.0	**41.8 ± 3.6**	1175
CIM	auth. type-shared with-name	**8.7 ± 3.2**	7.6 ± 3.0	226
CIM	auth. type-type	52.7 ± 4.4	**54.3 ± 5.3**	397
CIM	title-duration	44.4 ± 4.6	**48.0 ± 4.9**	374
CIM	title-name	**38.4 ± 3.3**	37.0 ± 3.7	1050
CIM	title-title holders-address	28.1 ± 2.8	**33.7 ± 3.2**	884
CIM	title-title holders-ID number	22.2 ± 9.3	**38.2 ± 3.4**	755
CIM	title-title holders-name	31.7 ± 2.9	**36.4 ± 1.5**	1094
CIM	title-title holders-nationality	22.6 ± 4.6	**35.4 ± 4.7**	215
CIM	title-valid from	0.0 ± 0.0	**6.2 ± 4.2**	33
CIM	title-valid until	4.3 ± 2.4	**24.8 ± 3.4**	398
CIM	Average	26.8 ± 3.6	**33.0 ± 3.7**	
CWC	change	**55.7 ± 4.5**	53.6 ± 5.3	384
CWC	change amount	37.3 ± 6.9	**41.3 ± 5.0**	296
CWC	new capital-amount	33.9 ± 2.3	**38.2 ± 4.2**	409
CWC	new capital-num. shares	44.0 ± 3.2	**48.5 ± 6.7**	301
CWC	new capital-share price	45.2 ± 3.6	**45.5 ± 5.0**	300
CWC	new capital-shares-amount	**42.2 ± 4.9**	39.7 ± 7.1	254
CWC	old capital-amount	37.3 ± 2.9	**37.5 ± 3.6**	230
CWC	Average	42.2 ± 4.0	**43.5 ± 5.3**	
CC	action	74.2 ± 3.6	**77.4 ± 2.1**	417
CC	duration	48.0 ± 12.0	**78.2 ± 3.4**	352
CC	term	62.9 ± 4.8	**71.1 ± 3.7**	415
CC	valid from	42.2 ± 17.3	**80.5 ± 4.0**	302
CC	Average	56.8 ± 9.4	**76.8 ± 3.3**	
NTC	address	58.4 ± 18.8	**90.6 ± 2.6**	493
NTC	ann. number	70.8 ± 14.6	**98.4 ± 0.6**	494
NTC	cause	92.6 ± 1.1	**93.9 ± 1.2**	494
NTC	duration	90.8 ± 1.7	**91.3 ± 1.1**	492
NTC	valid from	61.2 ± 17.3	**87.2 ± 4.1**	481
NTC	Average	74.8 ± 10.7	**92.3 ± 1.9**	
Overall		44.1 ± 5.9	**53.2 ± 3.7**	

Table 4.19 : Overall performance comparison when field-aware path expansion is applied in terms of micro F1 score.

Model	Doc2EDAG	Doc2EDAG + Field-aware Path Expansion	Doc2EDAG	Doc2EDAG + Field-aware Path Expansion
Input Type	gold	gold	pred	pred
Event				
CIM	**49.1 ± 4.8**	38.3 ± 5.2	**33.4 ± 3.4**	23.4 ± 2.4
CWC	52.8 ± 10.7	**53.3 ± 11.5**	**43.5 ± 3.7**	36.1 ± 1.6
CC	**94.2 ± 1.5**	90.0 ± 2.4	62.5 ± 6.8	**76.1 ± 1.7**
NTC	**99.4 ± 0.3**	99.3 ± 0.3	77.4 ± 8.8	**90.1 ± 2.9**
Average	**73.9 ± 4.3**	70.2 ± 4.9	54.2 ± 5.6	**56.4 ± 2.2**

to choose in each field. However, in more complex events like CIM and CWC, when the partial information is combined with filtering, the model seems to have a harder time matching arguments to fields. Both in CIM and CWC, all roles except one observes drastically decreased performance. In CIM, many events of the same type can appear in the same announcement. In CWC, arguments in different role have the same entity type. Thus, when irrelevant arguments are not filtered, learning what arguments to not match to a field seems to be good auxiliary task for the model. Despite the performance improvements in CC and NTC and the slight performance improvement overall, field-aware training is not suitable when faced with more complex events and is not a suitable change for Doc2EDAG. The performance over predicted arguments can be seen in Table 4.21. As with the case with gold arguments, the model's performance is worse in more complex event types and similar observations can be made.

Performance comparison of proposed changes with respect to Doc2EDAG over gold arguments and predicted arguments can be seen in Figure 4.2 and Figure 4.3. As highlighted in the figures, removing the CRF layer and transfer learning from event argument extraction experiments are proved to be the changes that improve the performance of Doc2EDAG when training on the TRG-EE dataset.

Table 4.20 : Comparison of F1 scores with gold arguments when field-aware path expansion is applied.

	Model	Doc2EDAG	Doc2EDAG + Field-aware Path Expansion	
	Input Type	gold	gold	Support
Event	Role			
CIM	action	**55.0 ± 3.9**	47.2 ± 3.0	1175
CIM	auth. type-shared with-name	**19.9 ± 5.1**	3.8 ± 2.1	226
CIM	auth. type-type	**68.3 ± 3.6**	56.8 ± 6.6	397
CIM	title-duration	**65.3 ± 4.8**	48.3 ± 6.1	374
CIM	title-name	**51.6 ± 4.8**	47.3 ± 2.8	1050
CIM	title-title holders-address	**45.0 ± 5.5**	29.3 ± 5.6	884
CIM	title-title holders-ID number	**39.7 ± 6.8**	30.2 ± 7.5	755
CIM	title-title holders-name	**46.1 ± 4.2**	26.8 ± 10.0	1094
CIM	title-title holders-nationality	**26.5 ± 5.3**	21.7 ± 7.1	215
CIM	title-valid from	**47.8 ± 11.2**	35.3 ± 12.9	33
CIM	title-valid until	***19.6 ± 4.3***	10.4 ± 2.5	398
CIM	Average	**44.1 ± 5.4**	32.5 ± 6.0	
CWC	change	55.8 ± 10.3	**66.8 ± 6.3**	384
CWC	change amount	50.0 ± 10.3	**53.7 ± 13.2**	296
CWC	new capital-amount	**50.0 ± 8.3**	45.6 ± 11.3	409
CWC	new capital-num. shares	**50.7 ± 9.7**	48.3 ± 11.0	301
CWC	new capital-share price	**48.7 ± 10.1**	45.8 ± 12.4	300
CWC	new capital-shares-amount	**49.1 ± 10.7**	46.6 ± 11.1	254
CWC	old capital-amount	**55.8 ± 7.1**	48.9 ± 12.4	230
CWC	Average	**51.5 ± 9.5**	50.8 ± 11.1	
CC	action	93.3 ± 0.0	**94.5 ± 1.5**	417
CC	duration	**94.1 ± 1.8**	89.1 ± 3.6	352
CC	term	**94.4 ± 1.4**	85.0 ± 3.3	415
CC	valid from	**94.8 ± 1.6**	87.0 ± 5.5	302
CC	Average	**94.1 ± 1.6**	88.9 ± 3.5	
NTC	address	99.4 ± 0.4	**99.7 ± 0.3**	493
NTC	ann. number	99.4 ± 0.5	**100.0 ± 0.0**	494
NTC	cause	**100.0 ± 0.0**	**100.0 ± 0.0**	494
NTC	duration	**98.6 ± 1.0**	98.5 ± 0.8	492
NTC	valid from	**99.7 ± 0.1**	98.5 ± 0.7	481
NTC	Average	**99.4 ± 0.4**	99.3 ± 0.3	
Overall		**63.6 ± 5.0**	58.0 ± 5.9	

Table 4.21 : Comparison of F1 scores with predicted arguments when field-aware path expansion is applied.

	Model	Doc2EDAG	Doc2EDAG + Field-aware Path Expansion	
	Input Type	pred	pred	Support
Event	Role			
CIM	action	**41.6 ± 2.0**	26.9 ± 1.9	1175
CIM	auth. type-shared with-name	**8.7 ± 3.2**	0.0 ± 0.0	226
CIM	auth. type-type	**52.7 ± 4.4**	36.2 ± 3.3	397
CIM	title-duration	**44.4 ± 4.6**	36.2 ± 3.4	374
CIM	title-name	**38.4 ± 3.3**	25.7 ± 3.0	1050
CIM	title-title holders-address	**28.1 ± 2.8**	22.1 ± 1.6	884
CIM	title-title holders-ID number	**22.2 ± 9.3**	18.4 ± 3.4	755
CIM	title-title holders-name	**31.7 ± 2.9**	21.1 ± 2.4	1094
CIM	title-title holders-nationality	**22.6 ± 4.6**	17.8 ± 3.1	215
CIM	title-valid from	**0.0 ± 0.0**	**0.0 ± 0.0**	33
CIM	title-valid until	4.3 ± 2.4	**5.3 ± 2.2**	398
CIM	Average	**26.8 ± 3.6**	19.1 ± 2.2	
CWC	change	**66.8 ± 6.3**	47.1 ± 2.5	384
CWC	change amount	**53.7 ± 13.2**	34.1 ± 3.8	296
CWC	new capital-amount	**45.6 ± 11.3**	29.7 ± 4.3	409
CWC	new capital-num. shares	**48.3 ± 11.0**	33.3 ± 4.9	301
CWC	new capital-share price	**45.8 ± 12.4**	45.4 ± 2.5	300
CWC	new capital-shares-amount	**46.6 ± 11.1**	27.4 ± 2.4	254
CWC	old capital-amount	**48.9 ± 12.4**	27.4 ± 4.2	230
CWC	Average	**42.2 ± 4.0**	34.9 ± 3.5	
CC	action	74.2 ± 3.6	**77.0 ± 1.5**	417
CC	duration	48.0 ± 12.0	**75.5 ± 1.7**	352
CC	term	62.9 ± 4.8	**70.4 ± 2.2**	415
CC	valid from	42.2 ± 17.3	**83.5 ± 1.6**	302
CC	Average	56.8 ± 9.4	**76.6 ± 1.7**	
NTC	address	58.4 ± 18.8	**89.6 ± 2.2**	493
NTC	ann. number	70.8 ± 14.6	**93.2 ± 4.6**	494
NTC	cause	92.6 ± 1.1	**92.7 ± 1.4**	494
NTC	duration	**90.8 ± 1.7**	88.7 ± 3.7	492
NTC	valid from	61.2 ± 17.3	**84.7 ± 2.3**	481
NTC	Average	74.8 ± 10.7	**89.8 ± 2.8**	
Overall		44.1 ± 5.9	**44.8 ± 2.6**	

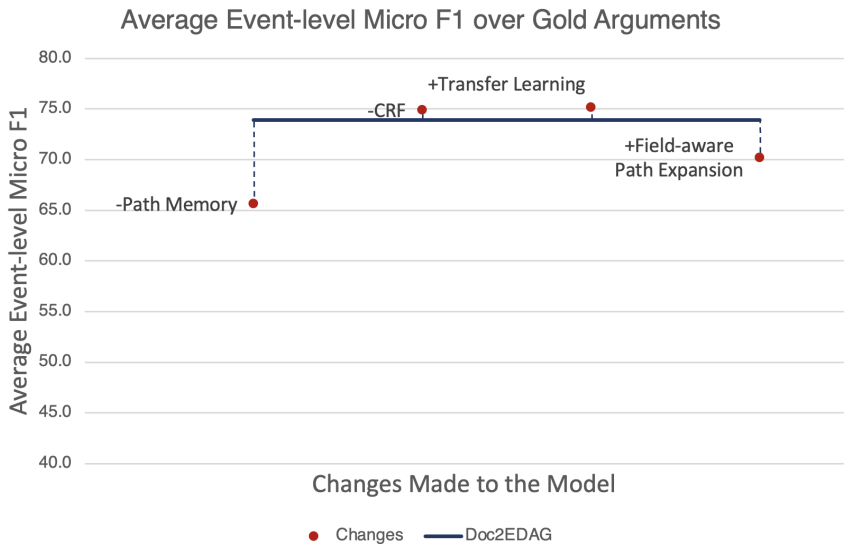

Figure 4.2 : Difference in performance provided by proposed changes with respect to Doc2EDAG in terms of average event-level micro F1 score over gold arguments.

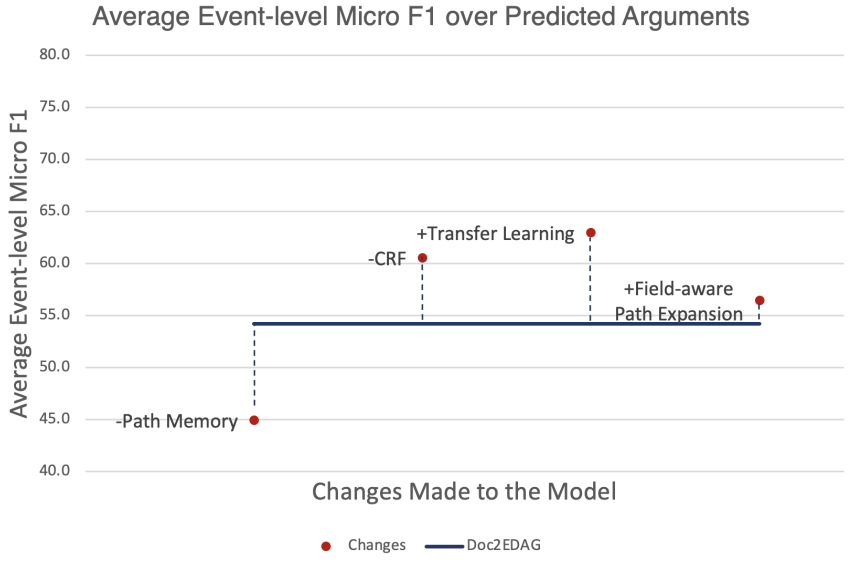

Figure 4.3 : Difference in performance provided by proposed changes with respect to Doc2EDAG in terms of average event-level micro F1 score over predicted arguments.

5. CONCLUSION

In this thesis, text data was extracted from the Turkish Trade Registry Gazette for information extraction. The Turkish Trade Registry Gazette if offered online for free for registered users. Since the issues are distributed in image PDF format, an image processing and OCR pipeline was implemented to extract text from pages.

Since the pages contain multiple arguments, the data was annotated for announcement boundaries and an text classification model was trained to detect boundaries. Using this model, the announcements were split and their types were determined by matching with metadata. With the matched metadata, an announcement classification dataset was produced. A BERT-based announcement classification model was trained on this dataset and the effect of context on prediction performance was observed. The model classifies announcements with an F1 score of 0.83.

The Turkish Trade Registry Gazette is an important source of information in many industries. Thus, an event extraction dataset was annotated. Four event categories were defined with their triggers and arguments. Different from existing financial event extraction datasets, this dataset also defines auxiliary entities. Auxiliary entities are complex entities consisting of triggers and arguments. 1284 announcements were manually annotated. To extract arguments and triggers, experiments were conducted. In these experiments, the effect of using IOB tags, adding a CRF layer and handling triggers and arguments separately were observed. The best performing model was determined to be the two-stage one that does not use IOB tags or a CRF layer, with a micro F1 score of 82.5.

For event extraction, a rule-based model and Doc2EDAG [1] was applied. The performances of the models were compared over both predicted arguments and gold arguments. In the setting where predicted arguments are used, the rule-based model used outputs of the two-stage model developed in the trigger and argument extraction experiments. Although the rule-based model performs better on simpler event types, it

fails as they get complicated. Doc2EDAG gave the best performance with an average micro F1 score of 73.9 on gold arguments and 54.2 on predicted arguments. Possible alterations on the model were also explored. Of these, removing the CRF layer and applying transfer learning yielded improved micro F1 scores of 74.9 and 75.2 over gold arguments and 60.5 and 62.9 over predicted arguments, respectively. The other two proposed methods, namely, turning off path expansion memory and field-aware path expansion yielded poorer results than the baseline.

As Doc2EDAG is quite resource-intensive, reductions were applied on the dataset. Future work can focus on increasing its processing capabilities.

REFERENCES

[1] **Zheng, S.**, **Cao, W.**, **Xu, W. and Bian, J.** (2019). Doc2EDAG: An End-to-End Document-level Framework for Chinese Financial Event Extraction, *EMNLP*.

[2] **Frisoni, G.**, **Moro, G. and Carbonaro, A.** (2021). A survey on event extraction for natural language understanding: Riding the biomedical literature wave, *IEEE Access*, *9*, 160721–160757.

[3] **Lv, J.**, **Zhang, Z.**, **Jin, L.**, **Li, S.**, **Li, X.**, **Xu, G. and Sun, X.** (2022). Trigger is Non-central: Jointly event extraction via label-aware representations with multi-task learning, *Knowledge-Based Systems*, *252*, 109480, `https://www.sciencedirect.com/science/article/pii/S0950705122007420`.

[4] **Sheng, J.**, **Guo, S.**, **Yu, B.**, **Li, Q.**, **Hei, Y.**, **Wang, L.**, **Liu, T. and Xu, H.** (2021). CasEE: A Joint Learning Framework with Cascade Decoding for Overlapping Event Extraction, `http://arxiv.org/abs/2107.01583`.

[5] **Gralinski, F.**, **Stanislawek, T.**, **Wróblewska, A.**, **Lipinski, D.**, **Kaliska, A.**, **Rosalska, P.**, **Topolski, B. and Biecek, P.** (2020). Kleister: A novel task for Information Extraction involving Long Documents with Complex Layout, *CoRR*, *abs/2003.02356*, `https://arxiv.org/abs/2003.02356`, `2003.02356`.

[6] **Zhou, Z.**, **Ma, L. and Liu, H.** (2021). Trade the Event: Corporate Events Detection for News-Based Event-Driven Trading, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Association for Computational Linguistics, Online, pp.2114–2124, `https://aclanthology.org/2021.findings-acl.186`.

[7] **Devlin, J.**, **Chang, M.W.**, **Lee, K. and Toutanova, K.** (2018). Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805*.

[8] **Lee, M.**, **Soon, L.K.**, **Siew, E.G. and Sugianto, L.F.** (2021). An annotated commodity news corpus for event extraction, *arXiv preprint arXiv:2105.08214*.

[9] **Lee, M.**, **Soon, L.K.**, **Siew, E.G. and Sugianto, L.F.** (2022). CrudeOilNews: An Annotated Crude Oil News Corpus for Event Extraction, *arXiv preprint arXiv:2204.03871*.

[10] **Liu, X.**, **Luo, Z. and Huang, H.** (2018). Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Brussels, Belgium, pp.1247–1256, https://aclanthology.org/D18-1156.

[11] **Strassel, S. and Mitchell, A.** (2003). Multilingual resources for entity extraction, *Proceedings of the ACL 2003 workshop on Multilingual and mixed-language named entity recognition*, pp.49–56.

[12] **Zhou, Y.**, **Chen, Y.**, **Zhao, J.**, **Wu, Y.**, **Xu, J. and Li, J.** (2021). What the Role is vs. What Plays the Role: Semi-Supervised Event Argument Extraction via Dual Question Answering, *35th AAAI Conference on Artificial Intelligence, AAAI 2021*, *16*, 14638–14646.

[13] **Huang, Y. and Jia, W.** (2021). Exploring Sentence Community for Document-Level Event Extraction, *Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021*, 340–351.

[14] **Yang, H.**, **Chen, Y.**, **Liu, K.**, **Xiao, Y. and Zhao, J.** (2018). DCFEE: A Document-level Chinese Financial Event Extraction System based on Automatically Labeled Training Data, *Proceedings of ACL 2018, System Demonstrations*, Association for Computational Linguistics, Melbourne, Australia, pp.50–55, https://aclanthology.org/P18-4009.

[15] **Zhu, T.**, **Qu, X.**, **Chen, W.**, **Wang, Z.**, **Huai, B.**, **Yuan, N. and Zhang, M.** (2022). Efficient Document-level Event Extraction via Pseudo-Trigger-aware Pruned Complete Graph, *1*, 4552–4558.

[16] **Guo, K.**, **Jiang, T. and Zhang, H.** (2020). Knowledge Graph Enhanced Event Extraction in Financial Documents, *2020 IEEE International Conference on Big Data (Big Data)*, pp.1322–1329.

[17] **Zheng, S.**, **Cao, W.**, **Xu, W. and Bian, J.** (2021). Revisiting the Evaluation of End-to-end Event Extraction, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 4609–4617.

[18] **Cui, S.**, **Cong, X.**, **Yu, B.**, **Liu, T.**, **Wang, Y. and Shi, J.** (2022). DOCUMENT-LEVEL EVENT EXTRACTION VIA HUMAN-LIKE READING PROCESS, 6337–6341.

[19] **Yang, H.**, **Sui, D.**, **Chen, Y.**, **Liu, K.**, **Zhao, J. and Wang, T.** (2021). Document-level event extraction via parallel prediction networks, *ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 6298–6308.

[20] **Xu, R.**, **Liu, T.**, **Li, L. and Chang, B.** (2021). Document-level Event Extraction via Heterogeneous Graph-based Interaction Model with a

Tracker, *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)*, Association for Computational Linguistics, Online, pp.3533–3546, https://aclanthology.org/2021.acl-long.274.

[21] **Liang, Y.**, **Jiang, Z.**, **Yin, D. and Ren, B.** (2022). *RAAT: Relation-Augmented Attention Transformer for Relation Modeling in Document-Level Event Extraction*, https://arxiv.org/abs/2206.03377.

[22] **Zhu, T.**, **Qu, X.**, **Chen, W.**, **Wang, Z.**, **Huai, B.**, **Yuan, N. and Zhang, M.** (2022). Efficient Document-level Event Extraction via Pseudo-Trigger-aware Pruned Complete Graph, *1*, 4552–4558.

[23] **Han, C.**, **Zhang, J.**, **Li, X.**, **Xu, G.**, **Peng, W. and Zeng, Z.** (2022). DuEE-Fin: A Large-Scale Dataset for Document-Level Event Extraction, ***W. Lu, S. Huang, Y. Hong and X. Zhou**, editors, Natural Language Processing and Chinese Computing*, Springer International Publishing, Cham, pp.172–183.

[24] **Li, X.**, **Li, F.**, **Pan, L.**, **Chen, Y.**, **Peng, W.**, **Wang, Q.**, **Lyu, Y. and Zhu, Y.** (2020). DuEE: a large-scale dataset for Chinese event extraction in real-world scenarios, *CCF International Conference on Natural Language Processing and Chinese Computing*, Springer, pp.534–545.

[25] **Adalı, K. and Tantuğ, A.C.** (2022). Annotation of Financial Entities Using A Comprehensive Scheme in Turkish, *2022 30th Signal Processing and Communications Applications Conference (SIU)*, pp.1–4.

[26] **Kaynak, K.Ş. and Tantuğ, A.C.** (2023). TFEEC: Turkish Financial Event Extraction Corpus, ***J.M. Machado, P. Chamoso, G. Hernández, G. Bocewicz, R. Loukanova, E. Jove, A.M. del Rey and M. Ricca**, editors, Distributed Computing and Artificial Intelligence, Special Sessions, 19th International Conference*, Springer International Publishing, Cham, pp.49–58.

[27] **Hürriyetoğlu, A.**, **Yörük, E.**, **Mutlu, O.**, **Duruşan, F.**, **Yoltar, C.**, **Yüret, D. and Gürel, B.** (2021). Cross-Context News Corpus for Protest Event-Related Knowledge Base Construction, *Data Intelligence*, *3*(2), 308–335, https://doi.org/10.1162/dint_a_00092, https://direct.mit.edu/dint/article-pdf/3/2/308/1963469/dint_a_00092.pdf.

[28] **Oral, B.**, **Emekligil, E.**, **Arslan, S. and Eryiğit, G.** (2020). Information extraction from text intensive and visually rich banking documents, *Information Processing & Management*, *57*(6), 102361.

[29] **Kay, A.** (2007). Tesseract: An Open-Source Optical Character Recognition Engine, *Linux J.*, *2007*(159), 2.

[30] **Smith, R.** (2016). *Tesseract Blends Old and New OCR Technology*, https://www.primaresearch.org/das2016/tutorials, international Workshop on Document Analysis Systems (DAS2016).

[31] **Vaswani, A.**, **Shazeer, N.**, **Parmar, N.**, **Uszkoreit, J.**, **Jones, L.**, **Gomez, A.N.**, **Kaiser, L.u. and Polosukhin, I.** (2017). Attention is All you Need, ***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., https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[32] **Dosovitskiy, A.**, **Beyer, L.**, **Kolesnikov, A.**, **Weissenborn, D.**, **Zhai, X.**, **Unterthiner, T.**, **Dehghani, M.**, **Minderer, M.**, **Heigold, G.**, **Gelly, S.**, **Uszkoreit, J. and Houlsby, N.** (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, https://openreview.net/forum?id=YicbFdNTTy.

[33] **Radford, A.**, **Kim, J.W.**, **Xu, T.**, **Brockman, G.**, **McLeavey, C. and Sutskever, I.** (2022). Robust speech recognition via large-scale weak supervision, *arXiv preprint arXiv:2212.04356*.

[34] **Wu, Y.**, **Schuster, M.**, **Chen, Z.**, **Le, Q.V.**, **Norouzi, M.**, **Macherey, W.**, **Krikun, M.**, **Cao, Y.**, **Gao, Q.**, **Macherey, K.** *et al.* (2016). Google's neural machine translation system: Bridging the gap between human and machine translation, *arXiv preprint arXiv:1609.08144*.

[35] **Zhu, Y.**, **Kiros, R.**, **Zemel, R.**, **Salakhutdinov, R.**, **Urtasun, R.**, **Torralba, A. and Fidler, S.** (2015). Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books, *The IEEE International Conference on Computer Vision (ICCV)*.

[36] **Wolf, T.**, **Debut, L.**, **Sanh, V.**, **Chaumond, J.**, **Delangue, C.**, **Moi, A.**, **Cistac, P.**, **Rault, T.**, **Louf, R.**, **Funtowicz, M.**, **Davison, J.**, **Shleifer, S.**, **von Platen, P.**, **Ma, C.**, **Jernite, Y.**, **Plu, J.**, **Xu, C.**, **Scao, T.L.**, **Gugger, S.**, **Drame, M.**, **Lhoest, Q. and Rush, A.M.** (2020). Transformers: State-of-the-Art Natural Language Processing, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, pp.38–45, https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[37] **Schweter, S.** (2020). *BERTurk - BERT models for Turkish*, https://doi.org/10.5281/zenodo.3770924.

[38] **Ortiz Suárez, P.J.**, **Sagot, B. and Romary, L.** (2019). Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures,

Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, Leibniz-Institut für Deutsche Sprache, Mannheim, pp.9 – 16, http://nbn-resolving.de/urn:nbn:de:bsz:mh39-90215.

[39] **Tiedemann, J.** (2012). Parallel Data, Tools and Interfaces in OPUS, ***N.C.C. Chair)***, ***K. Choukri***, ***T. Declerck***, ***M.U. Dogan***, ***B. Maegaard***, ***J. Mariani***, ***J. Odijk and S. Piperidis***, *editors, Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, European Language Resources Association (ELRA), Istanbul, Turkey.

[40] **Belval, E.** (2022). *pdf2image*, https://github.com/Belval/pdf2image.

[41] **Bradski, G.** (2000). The OpenCV Library, *Dr. Dobb's Journal of Software Tools*.

[42] **Klie, J.C.**, **Bugert, M.**, **Boullosa, B.**, **Eckart de Castilho, R. and Gurevych, I.** (2018). The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation, *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, Santa Fe, New Mexico, pp.5–9, https://www.aclweb.org/anthology/C18-2002.

[43] **INCEpTION, T.** *INCEpTION User Guide*, https://inception-project.github.io/releases/26.1/docs/user-guide.html#sect_webannotsv.

[44] **Demirtaş, I.N.**, **Arslan, S. and Eryiğit, G.** (2022). Classifying Turkish Trade Registry Gazette Announcements, *2022 7th International Conference on Computer Science and Engineering (UBMK)*, pp.204–209.

[45] **Consortium, L.D.** *et al.* (2005). *ACE (Automatic Content Extraction) English Annotation Guidelines for Events Version 5.4. 3 2005.07. 01*.

[46] **Xiang, W. and Wang, B.** (2019). A Survey of Event Extraction from Text, *IEEE Access*, *7*, 173111–173137.

[47] **Du, X. and Cardie, C.** (2020). Event extraction by answering (almost) natural questions, *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 671–683.

[48] **Sang, E.F. and De Meulder, F.** (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, *arXiv preprint cs/0306050*.

CURRICULUM VITAE

Name SURNAME: İrem Nur Demirtaş

EDUCATION:

- **B.Sc.:** 2020, Istanbul Technical University, Faculty of Computer and Informatics Engineering, Department of Computer Engineering

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2020 - 2022 Prometeia SpA, Machine Learning Engineer
- 2022 - Prometeia SpA, Senior Machine Learning Engineer

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Demirtaş, İ.**, Arslan, S., Eyiğit, G. (2022). Classifying Turkish Trade Registry Gazette Announcements. 2022 7th International Conference on Computer Science and Engineering (UBMK) (pp. 204-209).